



**TUGAS AKHIR - TE141599**

## **SISTEM KEAMANAN PADA LINGKUNGAN PONDOK PESANTREN MENGGUNAKAN RASPBERRY PI**

**NUR AHMAD SYAHID  
NRP 2212100701**

**Dosen Pembimbing  
Dr. Muhammad Rivai, S.T., M.T.  
Suwito, S.T., M.T.**

**JURUSAN TEKNIK Elektro  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
Surabaya 2016**



**UNDERGRADUATE THESES - 141599**

## **THE SECURITY SYSTEM OF ISLAMIC BOARDING SCHOOL USING RASPBERRY PI**

**NUR AHMAD SYAHID  
NRP 2212100701**

**Supervisor  
Dr. Muhammad Rivai, S.T., M.T.  
Suwito, S.T., M.T.**

**DEPARTMENT OF ELECTRICAL ENGINEERING  
FACULTY OF INDUSTRIAL TECHNOLOGY  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA 2016**

**SISTEM KEAMANAN PADA LINGKUNGAN PONDOK  
PESANTREN MENGGUNAKAN RASPBERRY PI**

**TUGAS AKHIR**

**Diajukan Guna Memenuhi Sebagian Persyaratan Untuk  
Memperoleh Gelar Sarjana Teknik Elektro**

**Pada  
Bidang Studi Teknik Sistem Pengaturan  
Jurusan Teknik Elektro  
Institut Teknologi Sepuluh Nopember**

**Menyetujui,**

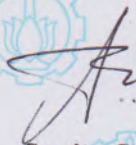
**Dosen Pembimbing I,**



**Dr. Muhammad Rivai S.T., M.T.**

**NIP:196904261994031003**

**Dosen Pembimbing II,**



**Suwito, S.T., M.T**

**NIP:198101052005011004**



## **SISTEM KEAMANAN PADA LINGKUNGAN PONDOK PESANTREN MENGGUNAKAN RASPBERRY PI**

**Nama Mahasiswa : Nur Ahmad Syahid**  
**NRP : 2212100701**  
**Jurusan : Teknik Elektro FTI-ITS**  
**Dosen Pembimbing 1 : Dr. Muhammad Rivai S.T., M.T.**  
**Dosen Pembimbing 2 : Suwito S.T., M.T**

### **ABSTRAK**

Salah satu pelanggaran kedisiplinan yang sering dilakukan oleh para santri ialah keluar dari pondok pesantren pada saat jam belajar pondok maupun saat tidak ada kegiatan pondok, oleh karena itu diperlukan alat bantu modern untuk membantu para ustad mengawasi lingkungan pondok pesantren. Dimana alat tersebut diimplementasikan dengan Raspberry Pi dan sensor PIR (*passive infrared*) serta modul kamera yang digunakan untuk mengawasi santri.

Sensor PIR digunakan untuk mendeteksi adanya gerakan. Setelah terdeteksi sebuah gerakan, modul kamera akan diaktifkan kemudian akan melihat dan mendeteksi objek menggunakan *histogram oriented of gradient* (HOG). Ketika objek terdeteksi adalah manusia maka objek itu difoto kemudian pesan pemberitahuan pelanggaran yang berisi gambar hasil foto dan pesan singkat akan dikirim melalui *email* ke *email* ustad.

Dari Uji coba sistem yang dibuat didapatkan hasil pendeteksian gerakan oleh sensor PIR dengan maksimal jarak 6 meter dan jarak pendeteksian oleh HOG berikisar antara 2 meter hingga 6 meter dan idealnya alat ini diletakkan 3 meter dari objek sehingga alat ini tepat digunakan untuk membantu meningkatkan penerapan ilmu kedisiplinan di lingkungan pondok pesantren.

***Kata kunci: Pondok Pesantren, Modul Kamera, Raspberry Pi, Sensor PIR, Histogram Oriented of Gradient .***

# **THE SECURITY SYSTEM OF ISLAMIC BOARDING SCHOOL USING REASPBERRY PI**

**Student's Name** : Nur Ahmad Syahid  
**Student's ID** : 2212100701  
**Department** : Teknik Elektro FTI-ITS  
**First Advisor** : Dr. Muhammad Rivai S.T., M.T.  
**Second Advisor** : Suwito S.T., M.T

## **ABSTRACT**

*One of the most undisciplined act that is frequently done by boarding school students is running away from the boarding school during the class or when there is no activity at the boarding school, because of that an modern helping device is needed to help the teachers to monitor the boarding school environment. The device will be implemented using Raspberry Pi, PIR (passive infrared) sensor, and camera modul in order to monitor the boarding school students.*

*PIR sensor is used to detect the movement. When a movement is detected, the camera module will be activated and it will soon catch and detect the object using Histogram Oriented of Gradient (HOG). If the detected object is a person, the object will be captured by the camera and a notification including the photo result and short message will be sent to the teacher's email.*

*The system testing showed that movement detection using PIR sensor result with a maximum distance of 6 meters and the detection distance using HOG is about 2 meters to 6 meters and this device is ideally placed 3 meters from the object so it can efficiently help the boarding school in improving the quality of school rules implementation.*

**Keywords:** *Boarding School, Camera Module, Raspberry Pi, PIR Sensor, Histogram Oriented of Gradient .*

## DAFTAR ISI

LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR .....	v
LEMBAR PENGESAHAN.....	vii
Abstrak .....	ix
<i>Abstract</i> .....	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR .....	xvii
DAFTAR TABEL .....	xix
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan .....	2
1.5 Manfaat .....	2
1.6 Metodologi .....	3
1.7 Sistematika Penulisan Laporan Tugas Akhir .....	4
BAB II TINJAUAN PUSTAKA.....	7
2.1 Raspberry Pi.....	7
2.2 <i>Socket</i> GPIO Raspberry Pi .....	9
2.3 HC-SR501 Passive Infrared Sensor (PIR) .....	10
2.4 Modul Kamera Raspberry Pi NoIR.....	12
2.5 Raspbian OS.....	13
2.6 SMTP ( <i>Simple Mail Transfer Protocol</i> ).....	14
2.7 Open CV .....	15
2.8 <i>Histogram of Oriented Gradients</i> .....	16
BAB III PERANCANGAN SISTEM .....	19
3.1 Deskripsi Sistem Secara Umum.....	19
3.2 Arsitektur umum Sistem .....	20
3.3 Perancangan Perangkat Keras .....	21
3.3.1 Implementasi Perangkat Keras .....	22
3.4 Perancangan Perangkat Lunak .....	25
3.4.1 Perancangan Isi Notifikasi .....	25
3.4.2 Perancangan Diagram Alir Data <i>Level 0</i> .....	26

BAB IV PENGUJIAN DAN EVALUASI .....	27
4.1 Lingkungan Uji Coba .....	27
4.2 Uji Coba Fungsionalitas Sistem .....	27
4.2.1 Uji Coba Fungsionalitas Pendeteksian Objek Bergerak dengan Sensor PIR Berdasarkan jarak .....	33
4.2.2 Uji Coba Fungsionalitas Pendeteksian Objek Manusia Dengan Kamera dan Fitur HOG Berdasarkan Jarak. ....	34
4.2.3 Uji Coba Fungsionalitas Pendeteksian Objek Manusia Dengan Kamera dan Fitur HOG Berdasarkan <i>Noise</i> . ....	36
4.2.4 Pengujian Perangkat Sistem.....	37
4.2.5 Skenario Uji Coba 1 .....	37
4.2.6 Skenario Uji Coba 2 .....	39
4.2.7 Skenario Uji Coba 3 .....	41
4.2.8 Skenario Uji Coba 4 .....	43
4.2.1 Skenario Uji Coba 5 .....	45
4.2.2 Skenario Uji Coba 6 .....	47
BAB V KESIMPULAN DAN SARAN .....	49
5.1 Kesimpulan.....	49
5.2 Saran.....	49
DAFTAR PUSTAKA.....	51
LAMPIRAN .....	53
A. Implementasi pada Raspberry Pi .....	53
B. Kode Program .....	57
C. Data Gambar Pengujian.....	60
BIODATA PENULIS.....	67

## DAFTAR TABEL

Tabel 4.1 Uji Coba Sensor PIR Berdasarkan Jarak.....	33
Tabel 4.2 Uji Coba Fitur HOG Berdasarkan Jarak.....	34
Tabel 4.3 Hasil Uji Coba 1.....	37
Tabel 4.4 Hasil Uji Coba 2.....	39
Tabel 4.5 Hasil Uji Coba 3.....	41
Tabel 4.6 Hasil Uji Coba 4.....	43
Tabel 4.7 Hasil Uji Coba 5.....	45
Tabel 4.8 Hasil Uji Coba 6.....	47



## DAFTAR GAMBAR

Gambar 2.1 Raspberry Pi 3 .....	8
Gambar 2.2 Fitur Raspberry Pi.....	9
Gambar 2.3 Pin GPIO Raspberry Pi.....	10
Gambar 2.4 Bagan Sensor HC-SR501 PIR .....	10
Gambar 2.5 Pancaran sensor PIR .....	11
Gambar 2.6 Modul Kamera Raspberry Pi NoIR .....	12
Gambar 2.7 Tampilan Raspberry OS Jessie .....	13
Gambar 2.8 <i>Syntax</i> SMTP objek .....	14
Gambar 2.9 R-HOG sel .....	18
Gambar 3.1 Alur diagram sistem .....	20
Gambar 3.2 Perangkat keras sistem pengawas.....	22
Gambar 3.3 Alat Pengawas Santri.....	22
Gambar 3.4 Rangkaian Perangkat Keras.....	24
Gambar 3.5 Rancangan isi notifikasi.....	25
Gambar 3.6 Diagram alir data level 0 .....	26
Gambar 4.1 Implementasi Sensor PIR .....	29
Gambar 4.2 Implementasi kamera dan HOG .....	30
Gambar 4.3 Kode Notifikasi dan fungsi pengiriman.....	32
Gambar 4.4 Hasil Implementasi Fungsi Kirim.....	32
Gambar 4.5 Uji Coba HOG Jarak 3 Meter .....	35
Gambar 4.6 Uji Coba Penambahan Noise .....	36
Gambar 4.7 Hasil dari Uji Coba 1 .....	38
Gambar 4.8 Laporan pada <i>email</i> dari Uji Coba 1.....	39
Gambar 4.9 Hasil Uji Coba 2 .....	40
Gambar 4.10 Laporan pada <i>email</i> dari Uji Coba 2.....	41
Gambar 4.11 Hasil Uji Coba 3 .....	42
Gambar 4.12 Laporan pada <i>email</i> dari Uji Coba 3.....	43
Gambar 4.13 Hasil Uji Coba 4 .....	44
Gambar 4.14 Laporan pada <i>email</i> dari Uji Coba 4.....	45
Gambar 4.15 Hasil Uji Coba 5 .....	46
Gambar 4.16 Laporan pada <i>email</i> dari Uji Coba 5.....	47
Gambar 4.17 Hasil Uji Coba 6 .....	48

# **BAB I**

## **PENDAHULUAN**

Pada bab ini akan dipaparkan mengenai garis besar tugas akhir yang meliputi latar belakang, tujuan, rumusan dan batasan permasalahan, metodologi pembuatan tugas akhir, dan sistematika penulisan.

### **1.1 Latar Belakang**

Seiring dengan perkembangan zaman pendidikan agama dinilai sebagai salah satu standar ilmu yang akhir akhir ini sering diterapkan sejajar dengan pendidikan formal. Pondok pesantren sebagai lembaga yang menerapkan pendidikan agama terpadu, sedangkan ilmu bidang lain dipandang sebagai alternatif yang menjanjikan untuk bentuk sekolah formal yang lain ada. Pondok pesantren juga membantu menerapkan pendidikan moral dan kedisiplinan, pesantren memberikan pendidikan moral yang bersinergi dengan pendidikan agama.

Namun, terkadang pendidikan kedisiplinan sulit diterapkan mengingat banyaknya santri dan sedikitnya sumber daya manusia. Untuk itu diperlukan sebuah alat bantu yang dapat membantu penerapan peraturan kedisiplinan. Salah satu peraturan pondok pesantren yang sering dilanggar oleh santri adalah keluar dari pondok pesantren pada saat jam belajar pondok, oleh karena itu diperlukan bantuan teknologi modern untuk membantu para ustad untuk mengawasi pondok.

Besar harapan penulis ini agar dapat membuat sistem untuk membantu para ustad mengawasi pondok pesantren dalam waktu 24 jam penuh dengan cara memanfaatkan teknologi Raspberry Pi, sensor PIR (*Passive Infra Red*) yang digunakan untuk mendeteksi adanya suatu gerakan, dan modul kamera yang digunakan untuk mengambil gambar dari suatu objek. Tugas Akhir diharapkan dapat menerapkan teknologi modern di lingkungan pondok pesantren yang dipandang oleh masyarakat luas bahwa pondok

pesantren masih menggunakan fasilitas atau sistem yang masih kuno atau tradisional.

## **1.2 Rumusan Masalah**

Rumusan masalah yang diangkat dalam Tugas Akhir ini adalah sebagai berikut:

1. Bagaimana membuat perangkat sistem yang dapat membantu ustad dalam hal pengawasan?
2. Bagaimana cara agar perangkat sistem dapat mendeteksi santri yang keluar melalui pagar pondok pesantren?
3. Bagaimana cara agar mendapatkan notifikasi dari perangkat sistem yang dibuat?

## **1.3 Batasan Masalah**

Beberapa batasan masalah yang menjadi batas pada Tugas Akhir ini adalah:

1. Perangkat sistem ini hanya mendeteksi objek manusia yang bergerak.
2. Objek manusia pada gambar harus utuh.
3. Notifikasi, dan gambar dikirim melalui *email*.

## **1.4 Tujuan**

Tujuan dari pembuatan Tugas Akhir ini adalah:

1. Dapat menerapkan kemajuan teknologi Raspberry Pi di lingkungan pondok pesantren.
2. Dapat membantu para ustad dalam hal menerapkan kedisiplinan pondok.
3. Laporan informasi pelanggaran ke ustad dapat bekerja.

## **1.5 Manfaat**

Tugas Akhir ini diharapkan mampu membantu meningkatkan keamanan dan kedisiplinan di lingkungan pondok pesantren dengan cara memanfaatkan teknologi Raspberry Pi dan

sensor yang sudah banyak dikembangkan di luar lingkungan pondok pesantren.

## 1.6 Metodologi

Tahapan-tahapan yang dilakukan dalam pengerjaan Tugas Akhir ini adalah sebagai berikut:

### 1. Penyusunan proposal Tugas Akhir.

Tahap pertama untuk memulai pengerjaan Tugas Akhir yaitu penyusunan proposal. Pada proposal ini, penulis mengajukan gagasan mengenai pemanfaatan teknologi Raspberry Pi, modul kamera Raspberry Pi, dan sensor PIR.

### 2. Studi literatur

Tahap ini merupakan tahap pengumpulan informasi yang diperlukan untuk pengerjaan Tugas Akhir sekaligus mempelajarinya. Mulai dari pengumpulan literatur, diskusi, serta pemahaman topik Tugas Akhir di antaranya tentang:

1. Perancangan perangkat sensor dari sistem pengawas perangkat keras Raspberry Pi.
2. Merancang sistem pengawas yang dapat bekerja dengan baik di lingkungan pondok pesantren.

### 3. Perancangan sistem

Pada tahap ini meliputi perancangan sistem berdasarkan studi literatur dan pembelajaran konsep sistem yang dibuat. Dengan bekal teori, metode dan informasi yang sudah terkumpul pada tahap sebelumnya diharapkan dapat membantu dalam proses tahap ini.

### 4. Implementasi perangkat lunak

Dalam pembuatan perangkat, digunakan beberapa teknologi untuk dapat menerapkan rancangan yang sudah ada, di antaranya:

- a. Raspbian

Merupakan sistem operasi Linux Debian yang dirancang untuk Raspberry Pi.

b. Bahasa Pemrograman

Sistem ini dibangun dengan menggunakan bahasa pemrograman Python. Penggunaan bahasa pemrograman diharapkan dapat membantu menangani kebutuhan sistem pada perangkat sensor untuk mendeteksi adanya objek manusia yang bergerak dan kebutuhan perangkat Raspberry Pi saat mengirimkan notifikasi, dan gambar ke *email*.

5. Pengujian dan evaluasi

Proses pengujian dilakukan di lingkungan pondok pesantren dengan cara memasang sensor pada tempat tertentu, sehingga apabila terjadinya suatu pelanggaran yang dilakukan oleh santri yaitu melewati pagar pondok pesantren maka sistem akan segera mendeteksi adanya pelanggaran.

6. Penyusunan buku Tugas Akhir.

Pada tahapan ini disusun buku yang memuat dokumentasi mengenai pembuatan serta hasil dari uji coba sistem yang telah dibuat.

## 1.7 Sistematika Penulisan Laporan Tugas Akhir

Buku Tugas Akhir ini disusun dengan sistematika penulisan sebagai berikut:

### **BAB I. PENDAHULUAN**

Bab yang berisi mengenai latar belakang, tujuan, dan manfaat dari pembuatan Tugas Akhir. Selain itu permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penulisan juga merupakan bagian dari bab ini.

## **BAB II. TINJAUAN PUSTAKA**

Bab ini berisi penjelasan secara detail mengenai dasar-dasar penunjang dan teori-teori yang digunakan untuk mendukung pembuatan Tugas Akhir ini.

## **BAB III. PERANCANGAN PERANGKAT LUNAK DAN KERAS**

Bab ini berisi tentang perancangan sistem, *flowchart*, dan perancangan sistem yang akan dibuat. Perancangan yang dibahas meliputi perancangan sistem, pengambilan gambar dan penghitungan jarak setelah terdeteksi oleh sensor infra merah, mengirimkan *email* yang berisi notifikasi, gambar, dan *map*.

## **BAB IV. IMPLEMENTASI**

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Penjelasan berupa *code* yang digunakan untuk proses implementasi.

## **BAB V. UJI COBA DAN EVALUASI**

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian fungsionalitas dan pengujian performa dalam beberapa skenario. Pengujian fungsionalitas merupakan pengujian jalannya aplikasi sesuai dengan perancangan sistem pada beberapa skenario yang telah ditentukan.

## **BAB VI. KESIMPULAN DAN SARAN**

Bab ini merupakan bab terakhir yang menyampaikan kesimpulan dari hasil uji coba yang dilakukan dan saran untuk pengembangan perangkat lunak ke depannya.

*[Halaman ini sengaja dikosongkan]*

## **BAB II**

### **TINJAUAN PUSTAKA**

Bab ini berisi penjelasan teori-teori yang berkaitan dengan implementasi sistem pengawas batas teritorial menggunakan Raspberry Pi, Penjelasan ini bertujuan untuk memberikan gambaran secara umum terhadap sistem yang dibuat. Tinjauan pustaka ini juga digunakan sebagai dasar yang menunjang pengembangan sistem pengawas ruangan menggunakan Raspberry Pi, sensor, modul kamera.

#### **2.1 Raspberry Pi**

Raspberry Pi adalah sebuah *Single Board Computer* seukuran kartu kredit yang dikembangkan oleh Raspberry Fondation dari UK dengan tujuan mempromosikan pengajaran ilmu komputer di sekolah dasar. Raspberry Pi menggunakan *system on a chip* (SoC) dari Broadcom BCM2835, dan juga termasuk prosesor ARM1176JZF-S 700 MHz, GPU VideoCore IV dan penyimpanan data jangka panjang. Meskipun mempunyai hampir semua kemampuan yang dimiliki komputer biasa, namun kemampuan komputasi Raspberry Pi tidak sekuat komputer pada umumnya. Aplikasi-aplikasi *open source* pun bisa dipasang ke dalam komputer mini tersebut seperti LibreOffice, *web browser* ataupun *programming* [1].

Beberapa sistem operasi yang dapat digunakan pada Raspberry Pi:

- Raspbian OS
- Arch Linux ARM
- Raspbmc
- OpenELEC

Pada gambar 2.1 menunjukkan bentuk fisik dari Raspberry Pi. Produksi Raspberry Pi sekarang diproduksi oleh China dan UK. Pada gambar 2.2 menunjukkan beberapa fitur yang disediakan dari Raspberry Pi yang terdiri dari *port* Lan, HDMI *output* untuk monitor, *power socket* sebagai sumber daya



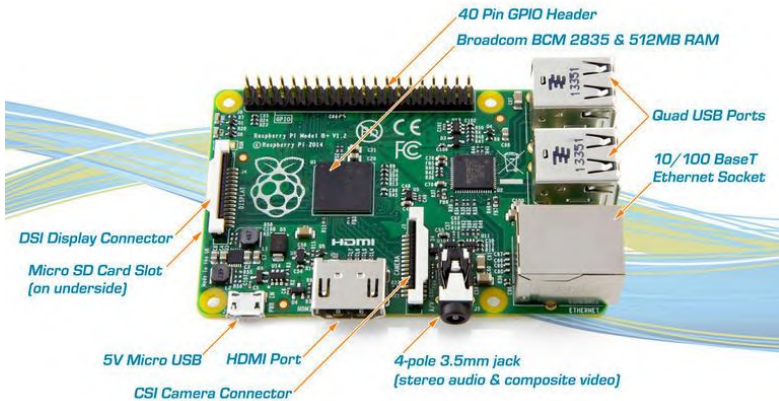
Raspberry Pi sebesar 5 volt 1 ampere, slot kartu micro SD yang mendukung sampai ukuran memori 32GB, *socket* GPIO yang digunakan untuk menghubungkan perangkat tambahan mikrokontroler, *socket* RCA video untuk tampilan ke TV *universal*, *socket* audio untuk luaran suara, lampu indikator *led* yang terdiri dari lampu indikator *on* dan *off*, indikator jaringan yang sedang berjalan, kemudian terdapat 2 buah *socket* USB yang dapat digunakan untuk berbagai macam kebutuhan seperti *keyboard*, *WIFI*, *mouse*, *flasdisk*, ataupun sebuah modul.

Fungsi lain dari perangkat Raspberry Pi sangat varian diantaranya yaitu:

- *Home Automation* yaitu suatu *control* pusat otomatis yang dapat dikendalikan untuk berbagai fungsi seperti halnya pintu otomatis, pengendali AC ataupun membuat CCTV.
- Dapat dikombinasikan dengan berbagai sensor seerti sensor cahaya, sensor gerak, sensor suhu dan lain lainnya.
- Dapat dijadikan sebagai *server Network Attached Storage*.
- Dapat dikombinasikan dengan perangkat Arduino.



Gambar 2.1 Raspberry Pi 3



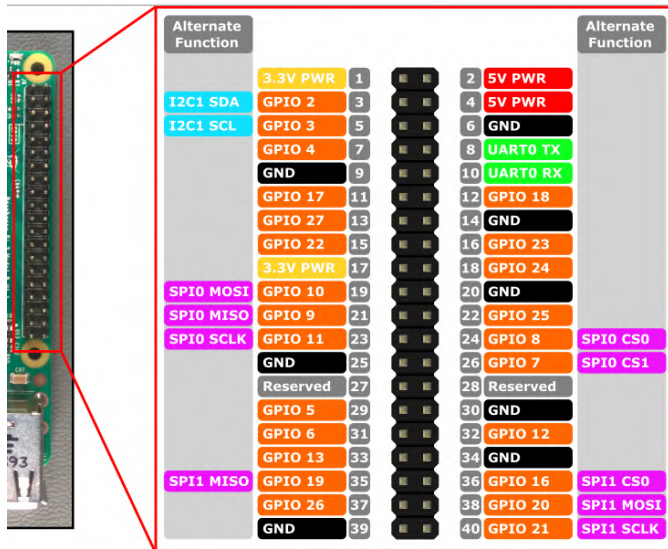
Gambar 2.2 Fitur Raspberry Pi

## 2.2 Socket GPIO Raspberry Pi

*General Purpose Input/Output* atau yang biasa disebut *socket GPIO* adalah pin yang dimana berfungsi sebagai *socket input/output* dari perangkat keras dan dapat dikendalikan melalui perangkat lunak. *Socket GPIO* dapat digunakan untuk menghubungkan beberapa perangkat tambahan seperti sensor infra merah, layar LCD, lampu LED ataupun untuk menambahkan perangkat Arduino. Kemampuan dari *socket GPIO* adalah antara lain sebagai berikut:

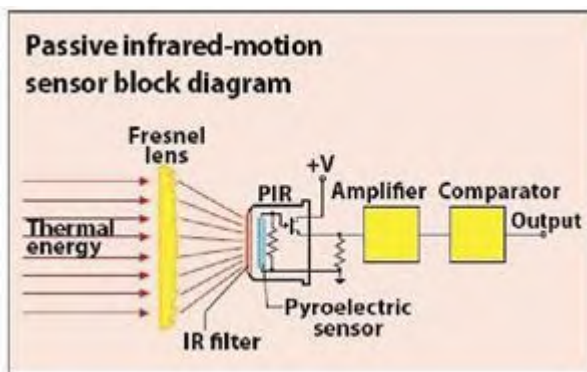
- Pin GPIO dapat diaktifkan dan di non aktifkan.
- Pin GPIO dapat dikonfigurasi untuk menerima *input* dan mengirim *output*.
- Nilai *input* dari GPIO adalah bernilai 1 (*high*) dan 0 (*low*).

Pada gambar 2.3 menunjukkan beberapa pin dari pin GPIO yang terdapat di Raspberry Pi, setiap pin memiliki kegunaanya masing-masing dan digunakan untuk berbagai perangkat tambahan [2].



Gambar 2.3 Pin GPIO Raspberry Pi

## 2.3 HC-SR501 Passive Infrared Sensor (PIR)



Gambar 2.4 Bagan Sensor HC-SR501 PIR

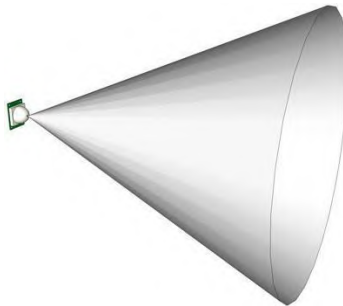
Pada gambar 2.4 menunjukkan proses kerja dari sensor PIR yaitu energi panas tubuh akan mempengaruhi resistansi

pyroelectric sensor. Resistansi yang berubah menyebabkan transistor keadaan aktif dan melawatkan arus dari common ke emitter dengan tegangan masukan 5v dan tegangan emiter ketika aktif adalah 3.3v

Sensor PIR bersifat pasif, artinya sensor ini tidak memancarkan sinar infra merah tetapi hanya menerima radiasi sinar infra merah dari luar, ketika sebuah gerakan terdeteksi dengan adanya suatu sumber radiasi dengan suhu tertentu yaitu manusia melewati sumber infra merah disekitarnya dengan suhu berbeda, maka sensor akan membandingkan pancaran infra merah yang diterima setiap satuan waktu, sehingga jika ada pergerakan maka akan terjadi perubahan pembacaan pada sensor. Tipe PIR yang khusus digunakan untuk mendeteksi manusia adalah tipe HC-SR501 *Body Sensor Module Pyroelectric Infrared* [3].

HC-SR501 *Body Sensor Module Pyroelectric Infrared* didasarkan pada teknologi inframerah dengan sensitivitas yang tinggi dan modus operasi *ultra-low-voltage*. Sensor ini memiliki *voltage* 5V-20V, *Power Consumption* 65mA, TTL output mencapai 3.3V, 0V. Suhu yang dideteksi memiliki temperature 20-70 derajat. jarak sensor bisa mencapai 5-7 meter dengan sudut pancaran mencapai kurang lebih 120 derajat [3].

Jarak pancar sensor PIR memiliki jangkauan jarak yang bervariasi, tergantung karakteristik sensor. Proses penginderaan sensor PIR dapat digambarkan sebagai berikut:



Gambar 2.5 Pancaran sensor PIR

Pada Gambar 2.5 ditunjukkan gambar bagaimana suatu sensor PIR mengeluarkan gelombangnya, pada umumnya sensor PIR memiliki jangkauan pembacaan efektif hingga 5 meter, dan sensor ini sangat efektif digunakan sebagai pendeteksi gerakan manusia.

## 2.4 Modul Kamera Raspberry Pi NoIR

Modul kamera Raspberry Pi NoIR adalah modul kamera yang dirancang khusus *add-on* untuk Raspberry Pi yang tidak memiliki *IR cut filter* yang dipasang seperti pada kamera biasa, dengan hanya memasangkan ke salah satu soket kecil di permukaan atas Raspberry Pi. *Interface* soket ini menggunakan CSI *interface* yang khusus, yang mana dirancang khusus untuk *interfacing* pada kamera.

Gambar 2.6 merupakan bentuk fisik dari modul kamera Raspberry Pi NoIR dimana modul tersebut memiliki ukuran sangat kecil, sekitar 25mm x 20mm x 9mm, dan juga memiliki berat hanya 3g. Modul kamera memiliki resolusi 5 megapixel, dan memiliki lensa fokus yang tetap pada *onboard*, kamera ini mampu mengambil gambar paling tinggi 2592 x 1944 pixel gambar statis, dan juga mendukung 1080p30, 720p60 dan 640 x 480p 60/90 video [4].

Versi NoIR adalah hal yang terbaik karena dapat digunakan pada malam hari, karena modul kamera Raspberry Pi NoIR dapat memancarkan cahaya inframerah.

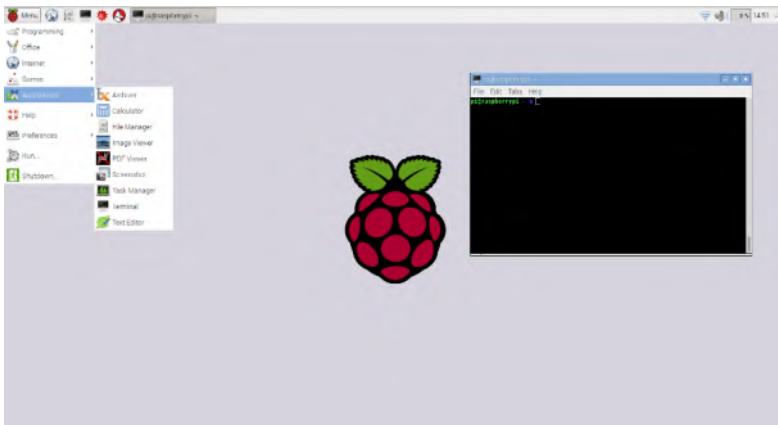


Gambar 2.6 Modul Kamera Raspberry Pi NoIR

## 2.5 Raspbian OS

Raspbian adalah sistem operasi Linux *distro* Debian yang dioptimalkan untuk penggunaan perangkat komputer mini Raspberry Pi. Sistem operasi ini memiliki beberapa *program* standard dan beberapa alat bantu untuk dapat menjalankan perangkat keras dari komputer mini Raspberry Pi. Di dalam sistem operasi Raspbian ini sudah lengkap daripada sistem operasi yang murni pada komputer umum, karena Raspbian berisi lebih dari 35.000 paket dan *pre-compiled* perangkat lunak yang tersaji dalam bentuk format yang mudah untuk diinstalasi dalam komputer mini Raspberry Pi [5].

Dengan 35.000 paket Raspbian, Raspbian sudah memiliki performa yang bagus yang dibangun dan telah selesai pada juni tahun 2012. Bagaimanapun, Raspbian pi masih dibawah developer aktif dan masih membutuhkan perkembangan yang jauh lebih baik, untuk paket paket lain memungkinkan masih bisa dibangun lagi. Gambar 2.7 di bawah merupakan salah satu contoh tampilan system operasi raspbian.



Gambar 2.7 Tampilan Raspberry OS Jessie

## 2.6 SMTP (*Simple Mail Transfer Protocol*)

*Simple Mail Transfer Protocol* atau SMTP adalah sebuah protocol yang dimana berfungsi untuk membantu dalam hal pengiriman *email* dan penjaluran pengiriman *email*. Dalam bahasa pemograman Python disediakan `smtplib` modul, yang dimana dapat menerjemahkan sebuah objek *client session* SMTP yang digunakan untuk melakukan pengiriman *email* ke semua perangkat internet yang mendukung pembacaan SMTP Daemon [6].

```
Import smtplib
smtpObj = smtplib.SMTP([host [, port [, local_hostname]]])
```

Gambar 2.8 *Syntax* SMTP objek

Pada Gambar 2.8 adalah sebuah *syntax* sederhana untuk membuat sebuah SMTP objek, yang dimana dapat digunakan untuk mengirim *email*.

Berikut ini adalah detail parameter Gambar 2.8

- Host: tempat dimana untuk menjalankan SMTP *server*, dapat juga digunakan sebagai penggunaan alamat IP dari sebuah *domain*.
- Port: jalur data yang digunakan untuk pertukaran data SMTP *server*.
- *Local\_hostname*: berisi *webmail service*, seperti contoh menggunakan google (`smtp.google.com`) atau *webmail* its (`smtp.its.ac.id`).

SMTP objek memiliki sebuah metode yang disebut “*sendmail*” yang digunakan untuk mengirimkan pesan. Terdapat 3 buah parameter yaitu:

- *Sender*: parameter yang berisi alamat dari pengirim.
- *Receiver*: parameter yang berisi alamat untuk penerima.
- *Message*: parameter yang berisi pesan dengan format tertentu.

## 2.7 Open CV

OpenCV adalah suatu *library* gratis yang dikembangkan oleh *developer-developer* Intel Corporation. *Library* ini terdiri dari fungsi-fungsi *computer vision* dan API (*Application Programming Interface*) untuk *image processing high level* maupun *low level* dan sebagai optimasi aplikasi *realtime*. OpenCV sangat disarankan untuk programmer yang akan berkecukupan pada bidang *computer vision*, karena *library*nya mampu membuat aplikasi yang handal dibidang *digital vision* dan mempunyai fitur yang mirip dengan cara pengolahan visual pada manusia [7].

Berikut ini adalah beberapa fitur pada *library* OpenCV:

- Manipulasi data gambar (alokasi memori, melepaskan memori, setting serta konversi gambar)
- Image/Video I/O (Bisa menggunakan kamera yang sudah didukung oleh *library* ini)
- Manipulasi matriks dan vektor serta terdapat juga *routines* linier algebra (*products, solvers, eigenvalues, SVD*)
- *Image processing* dasar (filtering, edge detection, pendeteksian tepi, sampling dan interpolasi, konversi warna, operasi morfologi, *histograms, image pyramids*)
- Analisis struktural
- Kalibrasi kamera
- Pendeteksian gerak
- Pengenalan objek (Hewan, Manusia, Tumbuhan dll)
- Basic GUI (*Display* gambar/video, mouse/ keyboard kontrol, *scrollbar*)
- *Image Labelling* (*line, conic, polygon, text drawing*)

Dengan memanfaatkan *library* pada openCV yaitu fungsi HOG (*Histogram Oriented of Gradient*) *descriptor*, maka suatu image dapat dideteksi objek tersebut atau manusia atau bukan. HOG *descriptor* sendiri merupakan suatu fitur dari gambar yang



digunakan untuk menghitung vektor gradien pada area tertentu sehingga dihasilkan output berupa vektor yang nantinya diklasifikasi oleh *support vector machine* [8].

## 2.8 *Histogram of Oriented Gradients*

Langkah pertama dalam metode histogram of oriented gradients adalah mengaplikasikan gambar yang dinormalisasi dengan pemerataan global yang opsional yang dirancang untuk mengurangi pengaruh efek pencahayaan. Yang digunakan adalah normalisasi menggunakan kompresi gamma, yaitu dengan cara menghitung akar kuadrat atau log dari setiap channel warna. Kekuatan tekstur gambar biasanya sebanding dengan pencahayaan permukaan lokal sehingga kompresi ini membantu mengurangi efek local shadow dan variasi iluminasi.

Langkah kedua adalah menghitung gradient dari gambar. Gradien dari suatu gambar dapat diperoleh dengan penyaringan dengan filter 2 dimensi yaitu filter vertikal dan horisontal. Yang pertama dilakukan adalah gambar dikonversi dalam bentuk grayscale untuk menghindari keharusan untuk mempertimbangkan kontribusi intensitas yang berbeda untuk setiap bidang warna (RGB). Metode yang biasa digunakan adalah 1-D centered, dengan matriks seperti berikut:

$$[-1, 0, 1]$$

$$1. \text{ Untuk sumbu } x : \frac{\partial f}{\partial x} = \frac{f(x+h) - f(x-h)}{2h} \quad (2.1)$$

$$2. \text{ Untuk sumbu } y : \frac{\partial f}{\partial y} = \frac{f(y+h) - f(y-h)}{2h} \quad (2.2)$$

Akan diperoleh nilai  $\frac{\partial f}{\partial x}$  dan  $\frac{\partial f}{\partial y}$  yang digunakan untuk menghitung *gradient*:

$$a. \text{ Magnitude (besar gradient): } R = \sqrt{x^2 + y^2}$$

$$b. \text{ Orientasi gradient (dalam sudut): } \theta = \arctan\left(\frac{y}{x}\right)$$

Langkah ketiga bertujuan untuk menghasilkan pengkodean

yang sensitif terhadap konten gambar lokal, namun tetap tahan terhadap perubahan kecil dalam pose dan penampilan. Gambar dibagi menjadi beberapa daerah dengan ruang kecil yang disebut sel. Untuk setiap sel akan diakumulasikan histogram 1-D lokal atau orientasi tepi pada semua pixel dalam sel. Kombinasi cell-level histogram 1-D membentuk representasi histogram orientasi dasar. Setiap histogram orientasi membagi berbagai sudut gradient menjadi angka tetap dalam bins yang telah ditentukan. Besarnya gradient dari pixel dalam sel digunakan untuk vote ke dalam histogram orientasi. Misalnya akan dibangun histogram yang didistribusikan melalui  $0^\circ$   $180^\circ$  dengan sejumlah channel sama dengan 9, maka untuk pemberian vote dalam histogram adalah sebagai berikut:

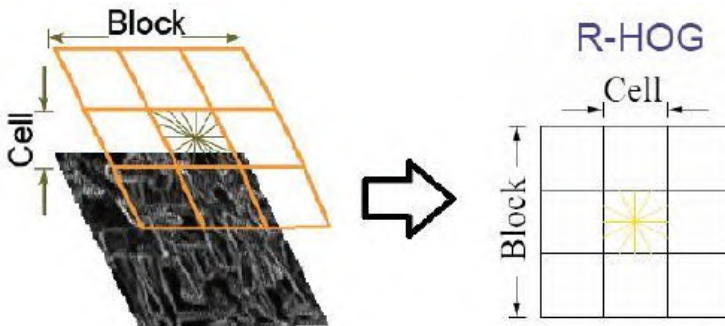
- a. Semua gradient dengan besar sudut  $[0^\circ-20^\circ]$  memberikan vote untuk channel 1.
- b. Semua gradient dengan besar sudut  $[20^\circ-40^\circ]$  memberikan vote untuk channel 2.
- c. Dan seterusnya.

Langkah keempat menghitung normalisasi, yang mengambil kelompok sel dan menormalisasi respon kontras secara keseluruhan. Hal ini dilakukan dengan mengakumulasi ukuran histogram dari grup sel yang disebut blok. Hasilnya akan digunakan untuk menormalisasi setiap sel dalam blok.

Cara menghitung normalisasi histogram dalam blok :

1. L1 – norm  $v \rightarrow v / (\|v\|_1 + \varepsilon)$ ;
2. L1 – sqrt:  $v \rightarrow \sqrt{v / (\|v\|_1 + \varepsilon)}$ ; -norm yang diikuti akar kuadrat, jumlah untuk memperlakukan *vector descriptor* sebagai distribusi peluang.
3. L2 – norm:
4. L2 – Hys,  $L2 - v \rightarrow v / \sqrt{\|v\|_2^2 + \varepsilon}$  *clipping* (dibatasi nilai maksimum sampai) dan renormalizing (Lowe,2004) dimana merupakan *vector* yang tidak dinormalisasi standar ke-k dan sebuah konstanta kecil.

Langkah kelima yaitu mengumpulkan descriptor dari semua blok yang merupakan overlapping grid yang mencakup detection window ke dalam feature vector gabungan untuk digunakan dalam classifier. Yang digunakan sebagai descriptor parameter series adalah R-HOG. Deskriptor blok R-HOG menggunakan sel berbentuk grid persegi atau persegi panjang yang overlap. R-HOG menghitung grid (yang mendefinisikan angka sel pada tiap blok) dari pixel sel yang masing-masing mengandung bins, dimana adalah parameter.



Gambar 2.9 R-HOG sel

Jadi apabila dengan asumsi memiliki:

1. Jendela ukuran
2. Sel-sel pixel (total ada sel)
3. Histogram 9 channel
4. Blok sel tanpa overlap (sebanyak 7 blok)

Akan diperoleh descriptor akhir ukuran:  $(7 \times 15) \times (2 \times 2) \times 9$   
 $= 3780$

## **BAB III**

### **PERANCANGAN SISTEM**

Perancangan merupakan bagian penting dari pembuatan perangkat lunak yang berupa perancangan-perancangan secara teknis mengenai sistem yang akan dibuat. Bab ini secara khusus akan menjelaskan perancangan sistem yang dibuat dalam Tugas Akhir ini. Penjelasan mencakup deskripsi umum aplikasi hingga perancangan proses, alur sistem.

#### **3.1 Deskripsi Sistem Secara Umum**

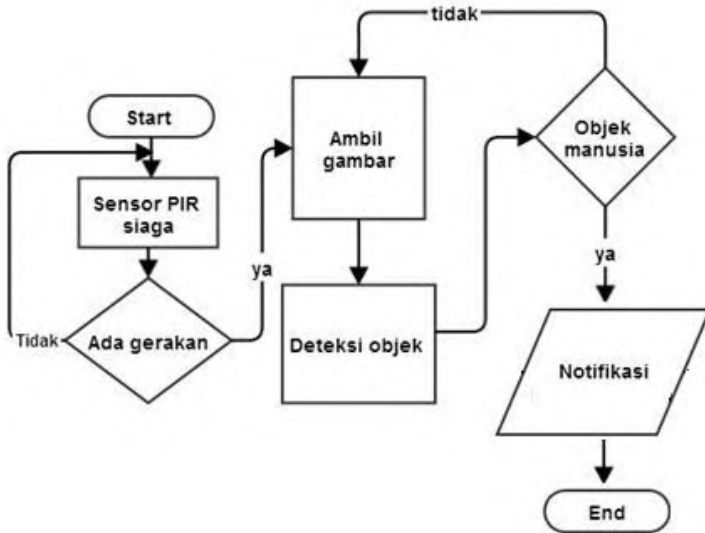
Pada Tugas Akhir ini dibangun suatu sistem pengawas batas teritorial pondok pesantren berbasis Raspberry Pi. Sistem ini dirancang dengan studi kasus untuk mengawasi batas teritorial pondok pesantren dari pelanggaran santri yang keluar pondok tanpa izin dan lemahnya pengawasan dari para ustad.

Sistem ini, pertama kali dilakukan ialah mengambil data dari input sensor infra merah yang dimana berfungsi sebagai penangkap suhu panas tubuh seseorang yang sedang keluar dari pondok pesantren yang sedang diawasi oleh rangkaian alat ini, sensor infra merah akan bernilai logika 0 dan 1, 0 saat sensor tidak mendeteksi adanya pancaran infra merah dan 1 saat sensor mendeteksi. Kemudian setelah sistem dapat mendeteksi adanya suatu gerakan maka modul kamera akan mendeteksi objek yang bergerak tersebut apakah manusia atau bukan setelah gambar objek bergerak tersebut diambil gambarnya. Setelah terdeteksi objek tersebut adalah manusia,

Untuk Raspberry Pi itu sendiri berfungsi sebagai pusat dari semua perangkat yang dihubungkan baik itu sensor infra merah, dan modul kamera.

### 3.2 Arsitektur umum Sistem

Rancangan arsitektur sistem ditunjukkan seperti pada gambar dibawah.



Gambar 3.1 Alur diagram sistem

Pada pada gambar 3.1 menunjukkan diagram alir dari proses kerja dari sistem pengawas menggunakan Raspberry Pi. Pertama proses inialisasi *start* kemudian mengaktifkan sensor PIR dan modul kamera dari Raspberry Pi agar sensor dan kamera pada posisi siaga, lalu jika terjadi gerakan maka sensor mendeteksi gerakan tersebut dan langsung memberikan sinyal ke Raspberry Pi agar modul kamera diaktifkan, setelah itu gambar dari kamera akan dideteksi menggunakan *HOG descriptor* agar dapat mengetahui apakah objek tersebut manusia atau bukan, jika manusia maka kamera melakukan perintah ambil gambar.. Kemudian Raspberry Pi mengirim notifikasi, dan gambar ke *email* tujuan.

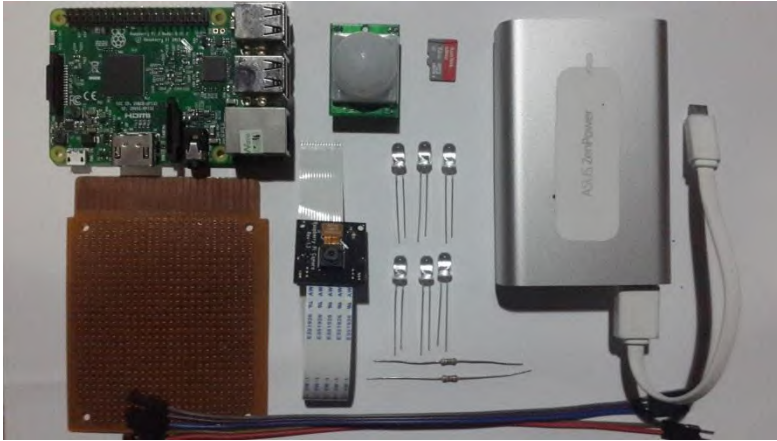
### 3.3 Perancangan Perangkat Keras

Perancangan perangkat keras pada penelitian ini menjelaskan tentang perangkat keras apa saja yang digunakan untuk mengimplementasikan Tugas Akhir ini. Tugas akhir ini membutuhkan perangkat keras sebagai berikut:

- 1 buah mini komputer Raspberry Pi 3 model B.
- 1 buah sensor PIR tipe HC-SR501
- 1 buah modul kamera Raspberry Pi NoIR
- 1 buah PCB
- 5 buah kabel pelangi *female to female*
- 1 buah *powerbank* oren 10050mah
- 1 buah SDHC *card* Sandisk Ultra 32GB
- 1 buah kabel konverter VGA ke HDMI

Pada gambar 3.2 menunjukkan beberapa kabel pelangi yang akan digunakan untuk menghubungkan *socket* GPIO ke sensor PIR dan modul kamera. Perangkat keras diatas juga menunjukkan Raspberry Pi sebagai mainboard atau sebagai pusat dari semua perangkat yang akan dihubungkan baik itu sensor PIR, dan modul kamera Raspberry Pi. Sensor PIR bertugas sebagai mendeteksi gerakan dari seseorang yang telah melewati rangkaian alat tersebut. Untuk *storage*, Raspberry Pi menggunakan SDHC *card* sebagai tempat penyimpanan berkas.

Seluruh perangkat sistem ini memakai daya suplai listrik 5 volt dan 2 ampere yang berasal dari *powerbank* berukuran 10600mah. Untuk perantara ke jaringan internet, perangkat sistem ini sudah memiliki modul *wifi* pada Raspberry Pi-nya agar dapat mengirimkan notifikasi berupa gambar dan teks melalui *email*.



Gambar 3.2 Perangkat keras sistem pengawas

### 3.3.1 Implementasi Perangkat Keras



Gambar 3.3 Alat Pengawas Santri

Pada subbab ini akan dijelaskan bagaimana implementasi perangkat keras pada Raspberry Pi agar semua perangkat keras dapat bekerja dengan baik dan sesuai dengan susunan yang ditentukan agar sistem pengawasan batas teritorial ini dapat berjalan dengan sempurna.

Pada gambar 3.3 menunjukkan hasil pembuatan perangkat keras yang akan digunakan pada sistem pengawasan pondok pesantren menggunakan Raspberry Pi, perangkat keras ini diawali dengan sebuah *prototype* untuk menguji apakah perangkat keras dapat berfungsi sebagai sistem pengawas pondok pesantren. berikut penjabaran dari perangkat keras yang digunakan:

- 1 buah mini komputer Raspberry Pi 3 model B.
- 1 buah sensor PIR tipe HC-SR501
- 1 buah sensor deteksi jarak HC-SR04
- 1 buah modul kamera Raspberry Pi NoIR
- 6 buah led *infrared* dan 2 resistor 22 ohm.
- sepotong PCB
- 11 buah kabel pelangi *male to female*
- 1 buah *powerbank* oren 10050mah
- 1 buah SDHC *card* Sandisk Ultra 32GB
- 1 buah *casing* akrilik persegi

Pengujian fungsi perangkat sensor melibatkan semua komponen perangkat keras. *Prototype* perangkat keras pada tugas akhir ini menggunakan sebuah komputer mini Raspberry Pi 3 model B yang dilengkapi dengan kartu SD berukuran 32GB dan sudah memiliki sistem operasi Raspbian Jessie, sensor PIR, modul kamera Raspberry Pi NoIR, dan komputer mini Raspberry Pi yang berguna sebagai pusat dari perangkat perangkat keras lainnya.

Pada gambar 3.4 menunjukkan rangkaian hubung antara komputer mini Raspberry Pi dengan sensor PIR dan modul kamera melalui *socket* GPIO raspberry Pi. Dibutuhkan beberapa kabel GPIO pelangi untuk menghubungkan *socket* dari sensor dan modul kamera ke Raspberry Pi melalui beberapa *socket* GPIO.

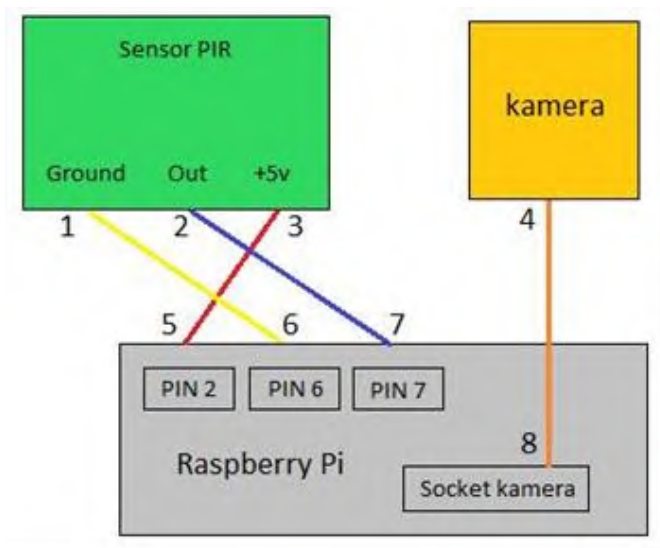
Pada sensor PIR terdapat tiga pin utama agar sensor dapat berjalan dengan baik. Ketiga pin tersebut ialah *vcc*, *ground*, dan *digital pin out*. Pin pertama pada sensor PIR yaitu *vcc* yang ditunjukkan pada nomor 3 di atas digunakan sebagai pin *power* yang harus diberi daya 5 volt maka ditancapkan pada Pin 2 yang



dimana Pin 2 merupakan sumber power 5 volt pada Raspberry Pi yang ditunjukkan oleh penomoran angka 5 seperti pada gambar di atas, lalu pin *ground* pada sensor PIR ditancapkan pada Pin 6 yang berfungsi sebagai *ground* pada Raspberry Pi pada penomoran 6 di atas dan terakhir yaitu pin *digital out* pada sensor PIR yang ditunjukkan pada nomor 2 akan dihubungkan ke Pin 7 yaitu GPIO 4.

Untuk pemasangan modul kamera Raspberry Pi NoIR dipasang tepat pada *socket* kamera Raspberry Pi pada nomor 8.

Rangkaian perangkat sistem ini diletakkan sekitar 1-2 meter dari kemungkinan tempat pelanggaran santri dan diletakkan diatas tembok, agar keseluruhan perangkat dapat bekerja dengan baik maka membutuhkan suplai daya listrik dari sebuah *powerbank* yang telah terisi dayanya agar tidak terjadi kekurangan daya pada saat uji coba. Dan memiliki *wifi modul* yang sudah ada pada Raspberry Pi sebagai koneksi internet.



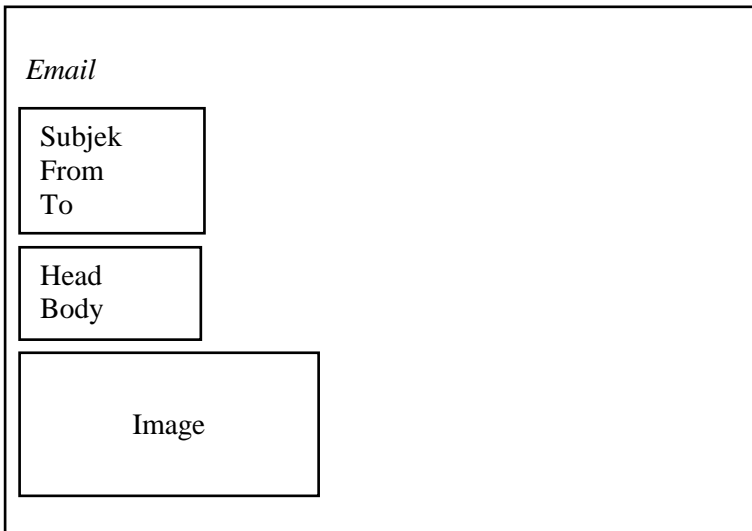
Gambar 3.4 Rangkaian Perangkat Keras

### 3.4 Perancangan Perangkat Lunak

Pada subbab ini akan dijelaskan bagaimana perancangan perangkat lunak pada Raspberry Pi agar perangkat keras dapat bekerja dengan baik dan sesuai dengan susunan yang ditentukan agar sistem pengawasan santri ini dapat berjalan dengan sempurna.

#### 3.4.1 Perancangan Isi Notifikasi

Perancangan ini merupakan perancangan antarmuka yang dimana mempunyai tujuan untuk memudahkan dalam memanfaatkan sistem ini dengan mudah. Rancangan antarmuka pada sistem ini hanya meliputi antarmuka hasil notifikasi yang akan diterima pengguna berupa isi notifikasi pada *email*. Rancangan antarmuka notifikasi ditunjukkan pada gambar 3.5 yang meliputi adanya subjek, pengirim, penerima, *head*, *body*, dan *file attachment* berupa gambar. Untuk isi notifikasi disusun menggunakan bahasa HTML agar terlihat rapi dan mudah dipahami oleh penerima notifikasi.



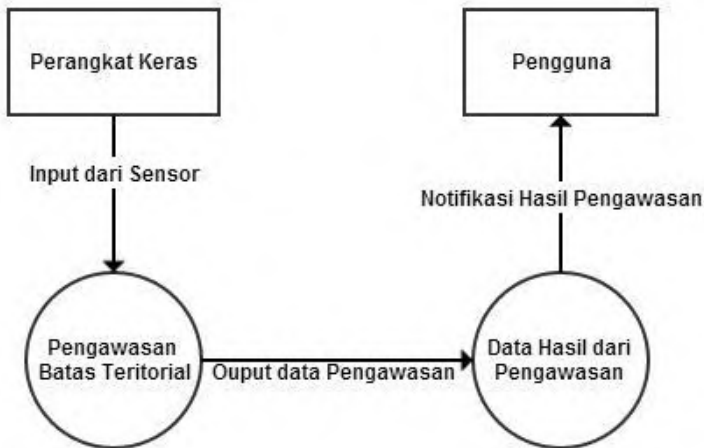
Gambar 3.5 Rancangan isi notifikasi

### 3.4.2 Perancangan Diagram Alir Data *Level 0*

Diagram alir data *level 0* menggambarkan fungsionalitas sistem beserta aktor yang terlibat. Seperti yang terlihat pada

Pada gambar 3.6 ditunjukkan bahwa terdapat satu pengguna yang menggunakan sistem ini. Pengguna akan menerima hasil dari pengawasan berupa notifikasi, dan gambar melalui *email* yang telah didaftarkan pada sistem ini.

Perangkat sistem ini diawali dengan *input* dari hasil pendeteksian gerak melalui sensor infra merah. Kemudian nilai dari *input* sensor infra merah akan men-*trigger* kamera yang terhubung dengan Raspberry Pi untuk melakukan perintah ambil gambar dan mendeteksi objek gambar tersebut. Setelah mendapatkan data informasi pengawasan, maka data tersebut dikirim ke pengguna melalui *email*.



Gambar 3.6 Diagram alir data level 0

## **BAB IV**

### **PENGUJIAN DAN EVALUASI**

Pada bab ini akan dijelaskan uji dari segi fungsionalitas dan performa dari sistem yang telah dirancang. Uji coba fungsionalitas dan performa akan dibagi ke dalam beberapa skenario uji coba.

#### **4.1 Lingkungan Uji Coba**

Pada subbab ini, dijelaskan mengenai gambaran dari lingkungan yang digunakan sebagai uji coba sistem. Uji coba akan dilakukan pada area suatu tembok atau pagar dan perangkat sistem diletakkan searah vertikal dengan tembok atau pagar, dimaksudkan agar dapat mendeteksi objek manusia yang akan melompat dari tembok. Pengiriman notifikasi akan secara langsung dikirim melalui *email* menggunakan koneksi internet dari *wifi* terdekat dengan fitur *wifi* modul yang ada pada Raspberry Pi. Lingkungan uji coba memiliki spesifikasi sebagai berikut:

- Laboratorium Elektronika B402.
- Rumah Keputih Tegal Bakti 2 No 3.
- Perangkat sistem pengawas pondok pesantren (Raspberry Pi, sensor PIR, modul kamera NoIR,)
- *Powerbank* 10600mah
- 1 buah Monitor Advance
- Laptop ASUS A450 Series Core 5
- 1 buah converter HDMI ke VGA

#### **4.2 Uji Coba Fungsionalitas Sistem**

Uji coba fungsionalitas merupakan sebuah pengujian terhadap jalannya fungsi–fungsi utama yang ada pada sistem. Pada sistem ini terdiri dari mini komputer Raspberry Pi, modul kamera dan sensor infra merah. Uji coba fungsionalitas ini

dilakukan pada ketiga bagian sistem tersebut. Fungsionalitas utama dari sistem ini terdapat pada mini komputer Raspberry Pi dan *script* bahasa pemrograman Python. Program dijalankan pada mini komputer Raspberry Pi dengan menggunakan *terminal* dari sistem operasi Raspbian Jessie. Ada tiga macam fungsionalitas utama yang merupakan bagian dari sistem ini yaitu:

- Pendeteksian sensor infra merah

Pendeteksian sensor infra merah merupakan salah satu bagian inti dari sistem karena sensor infra merah disini akan memiliki peran penting sebagai penerima sinyal panas tubuh manusia yang akan melewati suatu tembok yang akan dipantau oleh sistem ini, sensor infra merah akan menangkap gelombang yang dipancarkan oleh panas tubuh seseorang dan nantinya sensor yang terhubung langsung dengan mini komputer Raspberry Pi ini akan meneruskan perintah *live videostream* dan gambar akan ditangkap semuanya melalui modul kamera yang sebelumnya sudah di program menggunakan bahasa pemrograman Python.

Berikut implementasi program pada sensor PIR, fungsi sensor pada gambar 4.1 menunjukkan kondisi jika sensor mendeteksi gerakan, maka pada terminal akan tertulis “gerakan Terdeteksi”. Ketika tidak mendeteksi gerakan, maka pada terminal akan tertulis “Sensor PIR siap” yaitu singkatnya bernilai satu maka terdeteksi adanya suatu gerakan dan melakukan perintah selanjutnya, setelah itu sensor kembali ke mode siaga kembali.

```
#library untuk sensor PIR
import RPi.GPIO as GPIO;
import time;
GPIO.setmode(GPIO.BCM);
GPIO.setwarnings(False);
GPIO.setup(4,GPIO.IN);
previous = 0;
try:
    print "PIR Sensor is working!!";
    print "Ready to sense..";
    while True:
```

```

if GPIO.input(4)==1 and previous==0:
    print "Gerakan Terdeteksi";
    previous=1;
elif GPIO.input(4)==0 and previous==1:
    print "Sensor PIR siap";
    previous=0;
    time.sleep(0.01);
except KeyboardInterrupt:
    print "QUIT";
    GPIO.cleanup();

```

Gambar 4.1 Implementasi Sensor PIR

- *Streaming* kamera

Salah satu fitur dari sistem pengawas ruangan ini adalah fitur *live streaming* kamera dengan menampilkan *video* secara langsung dari modul kamera yang terhubung dengan mini komputer Raspberry Pi. Berikut implementasi program *Capture Image* dan *HOG Descriptor* pada modul kamera, proses mengambil gambar dari modul kamera Raspberry Pi NoIR dapat dijalankan melalui code seperti pada gambar 4.2, dengan menggunakan *Pivideostream* dapat secara mudah melakukan perintah *capture image* ketika objek manusia telah terdeteksi.

```

#library untuk kamera
from imutils.video.pivideostream import PiVideoStream
import cv2
from time import sleep
startcamera():
    cap = PiVideoStream(resolution = (240,180),framerate = 24)
    cap.start()
    hog = cv2.HOGDescriptor()
    hog.setSVMDetector(cv2.HOGDescriptor_
        getDefaultPeopleDetector())
    hogParams = {'winStride': (8, 8), 'padding': (32,
        32), 'scale': 1.05}

```

```

flag = 0
flagOld = 0
sleep(1)
while(1):
    frame = cap.read()
    (rects, result) = hog.detectMultiScale(frame,
        winStride=(4,4), padding=(8,8), scale=1.05)
    if result==():
        print result
    else:
        print "ada orang"
        print result
        cv2.imwrite('zz.jpg',frame)
        flag = 1
    if (flag == 0):
        if (flagOld == 1):
            kirimEmail()
    for (x, y, w, h) in rects:
        cv2.rectangle(frame,(x,y),(x+w, y+h),(0,255,0), 2)
    cv2.imshow("Camera",frame)
    key = cv2.waitKey(1) & 0xFF
    flagOld = flag

```

#### Gambar 4.2 Implementasi kamera dan HOG

Dengan menggunakan library openCV dan memanfaatkan fiturnya yaitu HOG descriptor [9] . Perangkat sistem ini dapat mendeteksi keberadaan objek manusia pada suatu gambar. Gambar 4.2 adalah code yang digunakan untuk mendeteksi suatu gambar dengan ukuran multi-scale windows.

- Pengiriman data gambar ke *email*

Pengiriman data gambar ke *email* merupakan bagian dari notifikasi yang dapat dilakukan oleh sistem kepada pengguna. Tujuan dilakukannya pengiriman data berupa gambar ke *email* pengguna adalah untuk memberi peringatan atau pemberitahuan kepada pengguna bahwa lokasi yang sedang dipantau oleh sistem

ini sedang di lalui oleh seseorang dan notifikasi ini akan berisi gambar yang nantinya akan disisipkan pada sebuah *email*. Berikut implementasi program notifikasi *email* dan pengiriman gambar. Gambar 3.8 menunjukkan fungsi kirim yang digunakan untuk mengirim hasil dari perangkat yang akan dikirim ke *email* sebagai notifikasi. Dengan menggunakan library *email* pada Python [10], parameter dari fungsi kirim ialah hasil dari informasi perangkat sensor dan modul yaitu pesan dan gambar. Pada gambar 3.8 adalah isi dari variabel `text_html`.

```
#library untuk kirim email
import os
import sys
from datetime import datetime
import smtplib
from email.mime.text import MIMEText
from email.mime.image import MIMEImage
from email.mime.multipart import
MIMEMultipart
kirimEmail():
strFrom =
'nur.ahmad.syahid12@mhs.ee.its.ac.id'
    strTo = 'nurahmadsyahid@gmail.com'
    msgRoot= MIMEMultipart('related')
    msgRoot['Subject']= 'Laporan
Pelanggaran'
    msgRoot['From'] = strFrom
    msgRoot['To']= strTo
    msgRoot.preamble = 'This is a multi-part
        message in MIME format'
    dt = str(datetime.now())
    text_html = '''<html>
                        <head>
                        </head>
                        <body><br>Assalamualaikum
                            Wr.Wb.<br/>
                            Telah terjadi pelanggaran
                            oleh santri<br/>
                            Terima Kasih<br/>
                        </body></html>'''
```



```

msgText = MIMEText (text_html, 'html')
msgRoot.attach(msgText)
fp = open('zz.jpg', 'rb')
img = MIMEImage(fp.read())
fp.close()
img.add_header('Content-ID', '<image>')
msgRoot.attach(img)
smtp = smtplib.SMTP('smtp.its.ac.id', 587)
smtp.starttls()
smtp.login('nur.ahmad.syahid12@mhs.ee.its.ac.id'
, 'Password Email')
smtp.sendmail(strFrom, strTo,
msgRoot.as_string())
smtp.quit()
print("Laporan Terkirim")

```

Gambar 4.3 Kode Notifikasi dan fungsi pengiriman  
 Pada Gambar 4.4 adalah hasil dari implementasi fungsi kirim berupa pesan *email*, yang terdiri dari pesan singkat dan informasi gambar.



Gambar 4.4 Hasil Implementasi Fungsi Kirim

#### 4.2.1 Uji Coba Fungsionalitas Pendeteksian Objek Bergerak dengan Sensor PIR Berdasarkan jarak

Pendeteksian objek bergerak menggunakan sensor PIR merupakan bagian inti dari sistem karena sensor PIR disini berperan penting sebagai penerima sinyal dari panas tubuh manusia yang akan melewati suatu tembok yang akan dipantau oleh perangkat sistem ini. Sensor PIR akan menangkap gelombang yang dipancarkan oleh panas tubuh seseorang. Pada uji coba ini sensor PIR di uji berdasarkan jarak, hasil uji coba ditunjukkan pada tabel 4.1

Tabel 4.1 Uji Coba Sensor PIR Berdasarkan Jarak

No.	Jarak	Keberhasilan
1	1 meter	Sukses
2	2 meter	Sukses
3	3 meter	Sukses
4	4 meter	Sukses
5	5 meter	Sukses
6	6 meter	Sukses
7	7 meter	Gagal
8	8 meter	Gagal
9	9 meter	Gagal
10	10 meter	Gagal

Dari tabel 4.1 di atas didapatkan bahwa jarak yang dapat dideteksi oleh sensor PIR adalah 1 meter hingga 6 meter.

#### 4.2.2 Uji Coba Fungsionalitas Pendeteksian Objek Manusia Dengan Kamera dan Fitur HOG Berdasarkan Jarak.

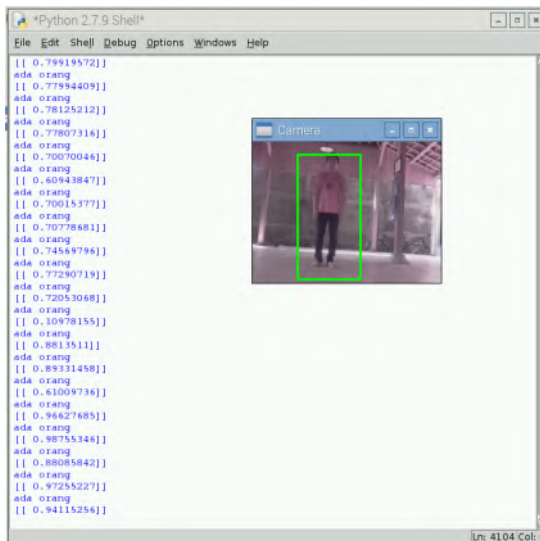
Pendeteksian objek menggunakan kamera dengan fitur HOG merupakan salah satu bagian inti dari sistem karena kamera dan HOG disini berperan penting sebagai penangkap dan pendeteksian objek manusia. Kamera akan menyala setelah sensor PIR mendeteksi adanya gerakan setelah itu objek akan dideteksi apakah itu manusia atau tidak menggunakan fitur HOG. Pada uji coba ini kamera dan HOG diletakkan 50 cm diatas permukaan tanah dan diuji daya tangkap berdasarkan jarak, hasil uji coba ditunjukkan pada tabel 4.2.

Tabel 4.2 Uji Coba Fitur HOG Berdasarkan Jarak

No.	Jarak	Keberhasilan
1	0 meter	Gagal
2	0.5 meter	Gagal
3	1 meter	Gagal
4	1.5 meter	Gagal
5	2 meter	Sukses
6	2.5 meter	Sukses
7	3 meter	Sukses
8	3.5 meter	Sukses
9	4 meter	Sukses
10	4.5 meter	Sukses
11	5 meter	Sukses

12	5.5 meter	Sukses
13	6 meter	Sukses
14	6.5 meter	Gagal
15	7 meter	Gagal

Dari tabel 4.2 di atas didapatkan bahwa objek pada jarak yang dapat dideteksi oleh HOG adalah 2 meter hingga 6 meter. Berikut gambar hasil uji coba kamera dan fitur HOG berdasarkan jarak pendeteksian:



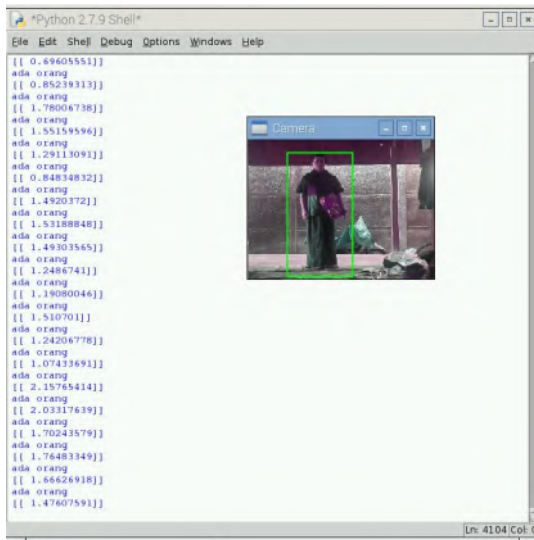
Gambar 4.5 Uji Coba HOG Jarak 3 Meter

Dari gambar 4.5 menunjukkan jarak ideal pendeteksian objek dan penempatan perangkat sistem pengawas yaitu 3 meter dari objek.

### 4.2.3 Uji Coba Fungsionalitas Pendeteksian Objek Manusia Dengan Kamera dan Fitur HOG Berdasarkan *Noise*.

Pada uji coba ini dilakukan pendeteksian objek manusia dengan memberikan beberapa *noise* berupa pemakaian atribut. Atribut yang dipakai adalah peci, sarung, dan sajadah yang dimana merupakan ciri khas dari santri pondok pesantren.

Pada gambar 4.6 menunjukkan kondisi seseorang dengan berbusana ala santri pondok pesantren. Dari gambar tersebut dapat diketahui bahwa pemakaian atribut tidak mempengaruhi pendeteksian objek oleh sistem ini.



Gambar 4.6 Uji Coba Penambahan Noise

## 4.2.4 Pengujian Perangkat Sistem

Pengujian perangkat sistem ialah uji coba yang dilakukan oleh semua bagian sistem yang sudah terhubung dan sudah berurutan jalannya perangkat sistem ini. Dimulai dengan sensor PIR yang mendeteksi objek bergerak tersebut, kemudian dilanjutkan ke modul kamera NoIR yang bertugas mengambil gambar dan mendeteksi objek pada gambar tersebut. dan yang terakhir informasi objek tersebut dikirim melalui *email* sebagai notifikasi.

### 4.2.5 Skenario Uji Coba 1

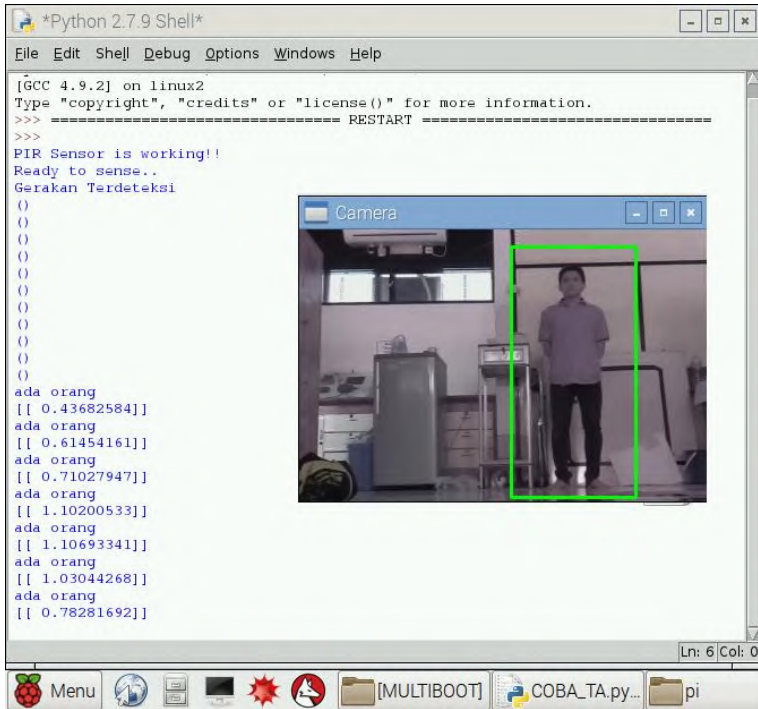
Skenario uji coba 1 adalah pengujian perangkat sistem di Laboratorium Elektronika Teknik Elektro. Uji coba ini untuk mendeteksi objek manusia saat menghadap ke depan.

Tabel 4.3 Hasil Uji Coba 1

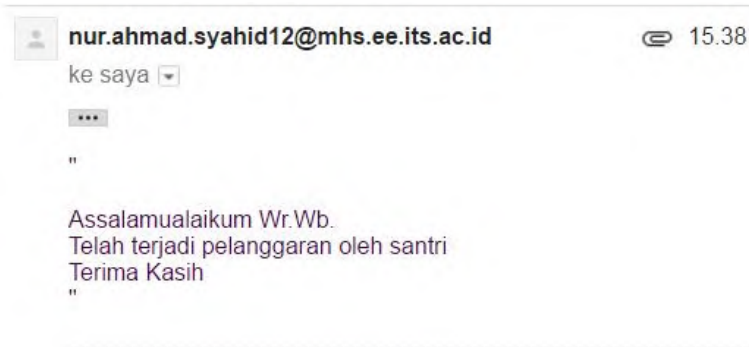
Posisi Objek Manusia	Menghadap ke depan
Sensor PIR	Bekerja
Kamera	Bekerja
HOG	Bekerja
Laporan	Terkirim
Tempat	Laboratorium Elektronika B402

Dari hasil uji coba pada tabel 4.3 dapat diketahui beberapa informasi dari perangkat sistem ini yaitu kondisi kerja sistem dan pada gambar 4.7 dan 4.8 menunjukkan proses kerja sistem dan hasil kirim informasi *email* berupa waktu dan gambar. Gambar dari hasil uji coba didapatkan dari hasil *capture* oleh modul kamera Raspberry Pi setelah sensor PIR bekerja dan mendeteksi

adanya suatu gerakan. Gambar menunjukkan adanya gerakan manusia.



Gambar 4.7 Hasil dari Uji Coba 1



Gambar 4.8 Laporan pada *email* dari Uji Coba 1

#### 4.2.6 Skenario Uji Coba 2

Skenario uji coba 2 adalah pengujian perangkat sistem di Laboratorium Elektronika Teknik Elektro. Uji coba ini untuk mendeteksi objek manusia saat menghadap ke samping kanan.

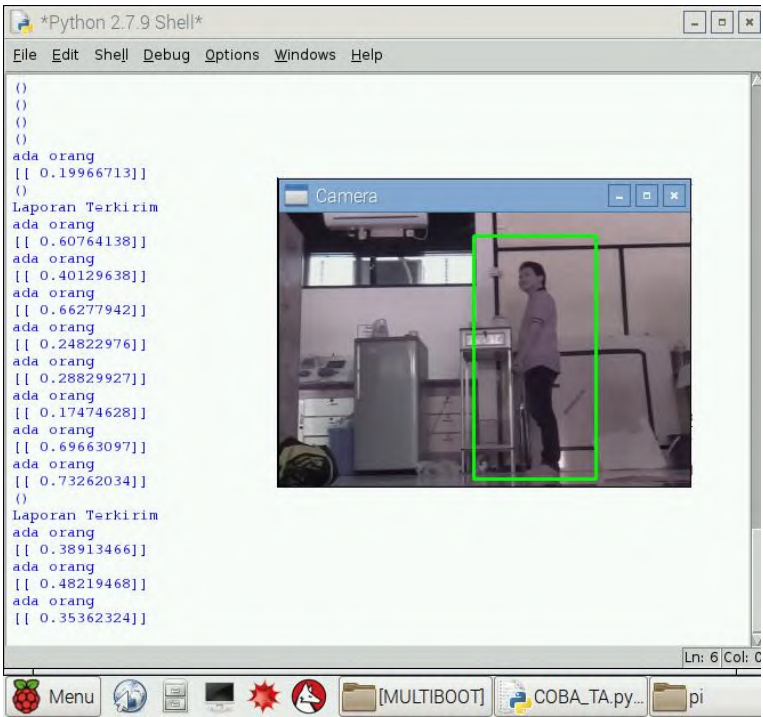
**Tabel 4.4 Hasil Uji Coba 2**

Posisi Objek Manusia	Menghadap ke kanan
Sensor PIR	Bekerja
Kamera	Bekerja



HOG	Bekerja
Laporan	Terkirim
Tempat	Laboratorium Elektronika B402

Dari hasil uji coba pada tabel 4.4 dapat diketahui beberapa informasi dari perangkat sistem ini yaitu kondisi kerja sistem dan pada gambar 4.9 dan 4.10 menunjukkan proses kerja sistem dan hasil kirim informasi *email* berupa waktu dan gambar.



Gambar 4.9 Hasil Uji Coba 2



Gambar 4.10 Laporan pada *email* dari Uji Coba 2

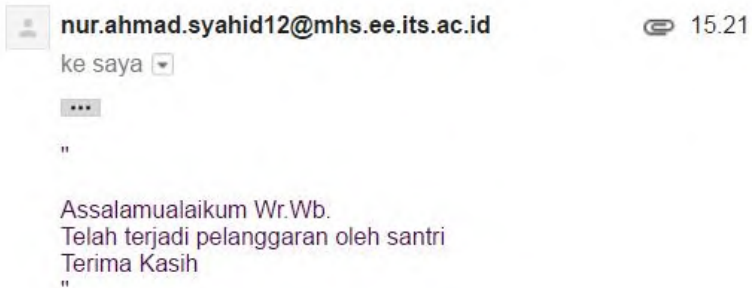
#### 4.2.7 Skenario Uji Coba 3

Skenario uji coba 3 adalah pengujian perangkat sistem di Laboratorium Elektronika Teknik Elektro. Uji coba ini untuk mendeteksi objek manusia saat menghadap ke samping kiri.

Tabel 4.5 Hasil Uji Coba 3

Posisi Objek Manusia	Menghadap ke kiri
Sensor PIR	Bekerja





Gambar 4.12 Laporan pada *email* dari Uji Coba 3

#### 4.2.8 Skenario Uji Coba 4

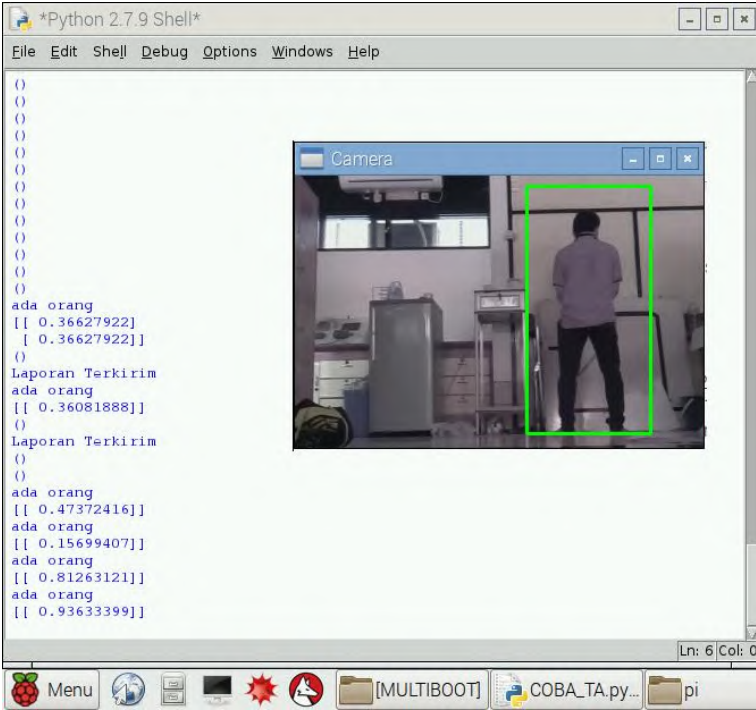
Skenario uji coba 4 adalah pengujian perangkat sistem di Laboratorium Elektronika Teknik Elektro. Uji coba ini untuk mendeteksi objek manusia saat menghadap ke belakang.

**Tabel 4.6 Hasil Uji Coba 4**

Posisi Objek Manusia	Menghadap ke belakang
Sensor PIR	Bekerja

Kamera	Bekerja
HOG	Bekerja
Laporan	Terkirim
Tempat	Laboratorium Elektronika B402

Dari hasil uji coba pada tabel 4.6 dapat diketahui beberapa informasi dari perangkat sistem ini yaitu kondisi kerja sistem dan pada gambar 4.13 dan 4.14 menunjukkan proses kerja sistem dan hasil fungsi kirim informasi *email* berupa waktu dan gambar.



Gambar 4.13 Hasil Uji Coba 4



Gambar 4.14 Laporan pada *email* dari Uji Coba 4

#### 4.2.1 Skenario Uji Coba 5

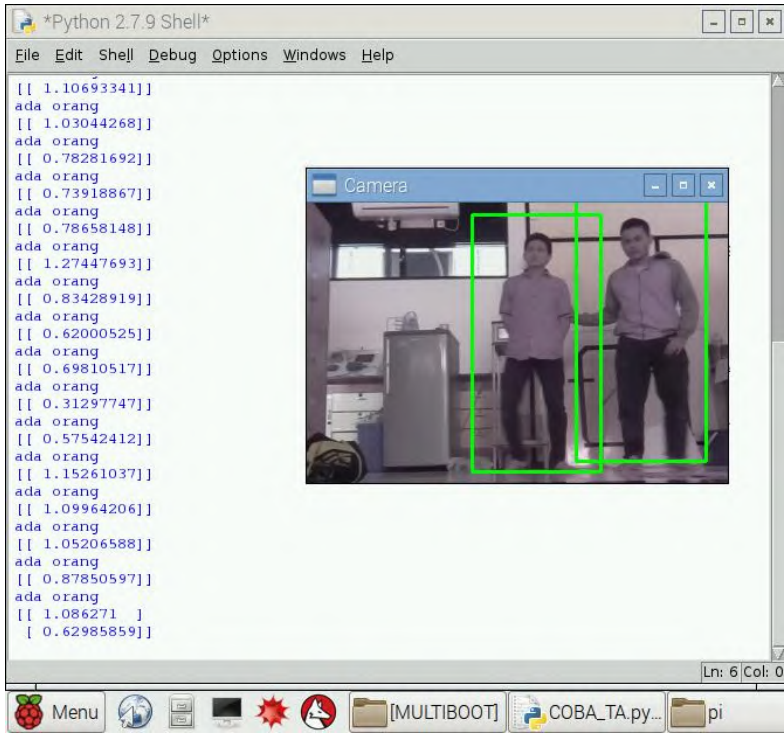
Skenario uji coba 5 adalah pengujian perangkat sistem di Laboratorium Elektronika Teknik Elektro. Uji coba ini dilakukan untuk mendeteksi jumlah manusia yang ditangkap oleh kamera. Diambil contoh 2 manusia.

Tabel 4.7 Hasil Uji Coba 5

Posisi Objek Manusia	Menghadap ke depan
Sensor PIR	Bekerja
Kamera	Bekerja

HOG	Bekerja
Laporan	Terkirim
Tempat	Laboratorium Elektronika B402

Dari hasil uji coba pada tabel 4.7 dapat diketahui beberapa informasi dari perangkat sistem ini yaitu kondisi kerja sistem dan pada gambar 4.15 menunjukkan proses kerja sistem yang dimana menguji berdasarkan jumlah manusia yang akan dideteksi dan gambar 4.16 menunjukkan hasil fungsi kirim informasi *email* berupa waktu dan gambar.



Gambar 4.15 Hasil Uji Coba 5



Gambar 4.16 Laporan pada *email* dari Uji Coba 5

#### 4.2.2 Skenario Uji Coba 6

Skenario uji coba 6 adalah pengujian perangkat sistem berdasarkan pendeteksian objek selain manusia bertempat di kontrakan saya sendiri. Uji coba ini dilakukan terhadap hewan yaitu kucing.

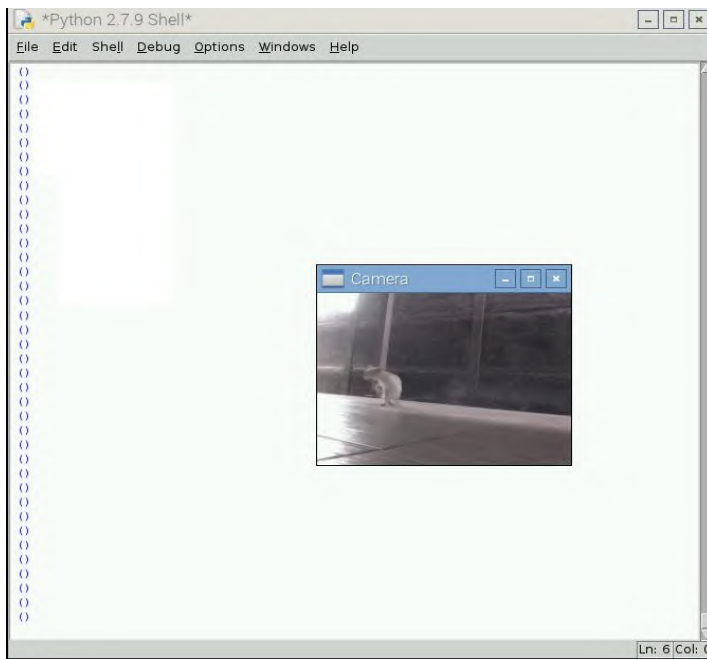
**Tabel 4.8 Hasil Uji Coba 6**

Posisi Objek Hewan	Menghadap ke depan
Sensor PIR	Bekerja
Kamera	Bekerja



HOG	Tidak Bekerja
Laporan	Tidak Terkirim
Tempat	Laboratorium Elektronika B402

Dari hasil uji coba pada tabel 4.8 dapat diketahui beberapa informasi dari perangkat sistem ini yaitu kondisi kerja sistem untuk mendeteksi apakah objek hewan dapat di tangkap oleh kamera atau tidak. Dan hasilnya gerakan hewan bisa terdeteksi namun sistem HOG tidak membaca sehingga fungsi *capture* kamera tidak bekerja yang ditunjukkan pada gambar 4.17 dan begitu pula pada laporannya tidak akan terkirim



Gambar 4.17 Hasil Uji Coba

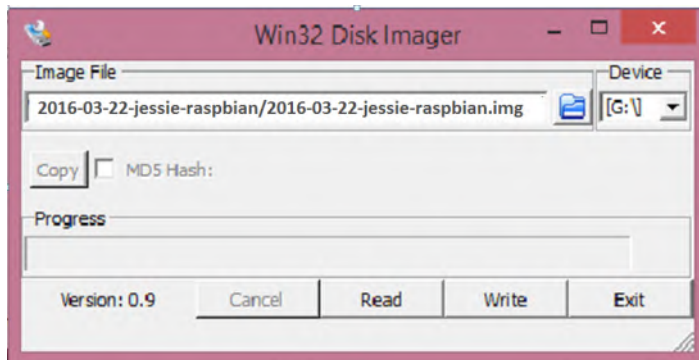
## LAMPIRAN

### A. Implementasi pada Raspberry Pi

- Memasang Raspbian Jessie

Proses instalasi sistem operasi pada Raspberry Pi adalah hal yang harus dilakukan dikarenakan Raspberry Pi seperti halnya suatu perangkat komputer perlu dipasang sistem operasi. Langkah- langkah untuk melakukan instalasi sistem operasi Raspbian Jessie akan dijelaskan ke beberapa tahapan sebagai berikut:

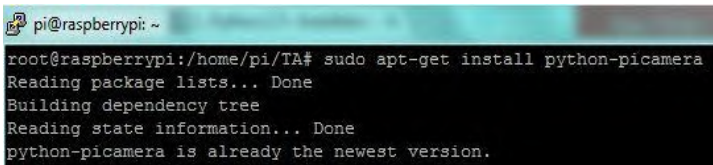
1. Siapkan sebuah kartu SD yang berukuran minimal 32GB.
2. Unduh aplikasi *Win32 Disk Imager* untuk melakukan format kartu SD yang akan digunakan sebagai wadah *file system* operasi Raspbian Jessie.



**Gambar A.1 Aplikasi Win32**

3. Pada Gambar A.1 ditunjukkan tampilan aplikasi *Win32 Disk Manager*, pilih *drive* kartu SD yang ingin di format dan sorot berkas *image* dari sistem operasi Raspbian Jessie.

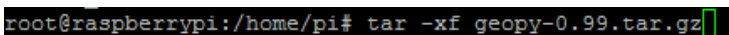
4. Klik *Write* dan tunggu sampai proses nya selesai dan kartu SD siap digunakan untuk menjalankan sistem operasi Raspberry Pi.
- Memasang *library* picamera  
 Proses instalasi *library* picamera digunakan agar modul kamera dari Raspberry Pi dapat melakukan perintah pengambilan gambar. Cara *install library* picamera sangat mudah dengan perintah pada Gambar A.2.



```
pi@raspberrypi: ~
root@raspberrypi:/home/pi/TA# sudo apt-get install python-picamera
Reading package lists... Done
Building dependency tree
Reading state information... Done
python-picamera is already the newest version.
```

**Gambar A.2 Install Library Picamera**

- Memasang *library* geopy 0.99  
 Hal yang pertama dilakukan dalam proses instalasi *library* geopy ialah mengunduh *package* geopy 0.99 di <https://pypi.python.org/pypi/geopy>.

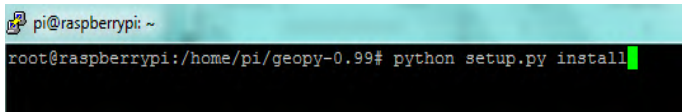


```
root@raspberrypi:/home/pi# tar -xf geopy-0.99.tar.gz
```

**Gambar A.3 Extract File Geopy**

Kemudian hasil unduhan di *extract* dengan perintah pada Gambar A.3 dan hasil *extract* ialah berupa folder geopy-0.99.

Masuk ke folder geopy-0.99 lalu lakukan perintah seperti pada Gambar A.4.



```
pi@raspberrypi: ~
root@raspberrypi:/home/pi/geopy-0.99# python setup.py install
```

**Gambar A.4 Install Geopy**

- Memasang OpenCV  
Langkah langkah *install library* OpenCV pada Raspberry Pi:
1. Memastikan sistem operasi Raspbian telah *update*:

```
sudo apt-get update
sudo apt-get upgrade
```

**Gambar A.5 Perintah Update dan Upgrade**

2. *Install* dependensi  
Pada Gambar A.6 dan Gambar A.7 adalah perintah *install* dependensi.

```
sudo apt-get -y install build-essential cmake cmake-curses-gui pkg-config
libpng12-0 libpng12-dev libpng++-dev libpng3 libpnglite-dev
zlib1g-dbq zlib1g zlib1g-dev pngtools
```

**Gambar A.6 Perintah Install Dependensi**

```

sudo apt-get -y install libjpeg8
libjpeg8-dev libjpeg8-dbg libjpeg-progs
ffmpeg libavcodec-dev libavcodec53
libavformat53 libavformat-dev
libgstreamer0.10-0-dbg
libgstreamer0.10-0 libgstreamer0.10-dev
libxine1-ffmpeg libxine-dev libxine1-
bin libunicap2 libunicap2-dev swig
libv4l-0 libv4l-dev python-numpy
libpython2.6 python-dev python2.6-dev
libqt2.0-dev

```

**Gambar A.7 Perintah *Install* Dependensi**

3. Kemudian *download library* OpenCV dari URL <http://opencv.org/downloads.html> , kemudian di *unzip file* *opencv-2.4.8.zip*. Lalu buat folder *release* atau *build* dengan perintah *mkdir release* atau *build* kemudian masuk ke folder tersebut.
4. Yang terakhir dalam proses instalasi ialah membuat perintah pada Gambar A.8:

```

make
sudo make install

```

**Gambar A.8 Perintah *Build* OpenCV**

- Memasang Sysstat  
Hal yang pertama dilakukan pertama kali ialah mengunduh paket *sysstat* dengan perintah seperti pada Gambar A.9:

```

Wget http://pagesperso-
orange.fr/sebastien.godard/sysstat-

```

**Gambar A.9 Unduh Paket Sysstat**

Kemudian ekstrak hasil dari unduh dengan perintah *tar xfv sysstat-11.0.0.tar.gz*, kemudian masuk ke folder hasil ekstrak *sysstat-11.0.0*. Lalu gunakan perintah seperti pada Gambar A.10 untuk melakukan instalasi.

```
./configure --enable-install-cron
```

**Gambar A.10 Perintah *Install Sysstat***

## B. Kode Program

- Fungsi menjalankan alat

```
#library untuk kirim email
import os
import sys
from datetime import datetime
import smtplib
from email.mime.text import MIMEText
from email.mime.image import MIMEImage
from email.mime.multipart import MIMEMultipart
#library untuk kamera
from imutils.video.pivideostream import
PiVideoStream
import cv2
from time import sleep
#library untuk sensor PIR
import RPi.GPIO as GPIO;
import time;
GPIO.setmode(GPIO.BCM);
GPIO.setwarnings(False);
GPIO.setup(4,GPIO.IN);
previous = 0;
def kirimEmail():
    strFrom =
'nur.ahmad.syahid12@mhs.ee.its.ac.id'
    strTo = 'nurahmadsyahid@gmail.com'
    msgRoot= MIMEMultipart('related')
    msgRoot['Subject']= 'Laporan Pelanggaran'
    msgRoot['From'] = strFrom
    msgRoot['To']= strTo
    msgRoot.preamble ='This is a multi-part
message in MIME format'
    dt = str(datetime.now())
```

```

        text_html = '''<html>
                        <head>
                        </head>
                        <body><br>Assalamualaikum
Wr.Wb.<br/>
                                Telah terjadi
pelanggaran oleh santri<br/>
                                Terima Kasih<br/>
                        </body></html>'''
        msgText = MIMEText (text_html,'html')
        msgRoot.attach(msgText)
        fp = open('zz.jpg', 'rb')
        img = MIMEImage(fp.read())
        fp.close()
        img.add_header('Content-ID', '<image>')
        msgRoot.attach(img)
        smtp = smtplib.SMTP('smtp.its.ac.id', 587)
        smtp.starttls()

smtp.login('nur.ahmad.syahid12@mhs.ee.its.ac.id',
'bismillah612nas')
        smtp.sendmail(strFrom, strTo,
msgRoot.as_string())
        smtp.quit()
        print("Laporan Terkirim")

def startcamera():
    cap = PiVideoStream(resolution =
(240,180),framerate = 24)
    cap.start()

    hog = cv2.HOGDescriptor()

hog.setSVMDetector(cv2.HOGDescriptor_getDefaultPeop
leDetector())
    hogParams = {'winStride': (8, 8), 'padding':
(32, 32), 'scale': 1.05}
    flag = 0
    flagOld = 0

    sleep(1)

    while(1):
        frame = cap.read()
        (rects, result) =
hog.detectMultiScale(frame, winStride=(4,4),
padding=(8,8), scale=1.05)
        if result==():
            print result
            flag = 0

```

```

else:
    print "ada orang"
    print result
    cv2.imwrite('zz.jpg',frame)
    flag = 1
    if (flag == 0):
        if (flagOld == 1):
            kirimEmail()

    for (x, y, w, h) in rects:
        cv2.rectangle(frame, (x,y), (x+w,
y+h), (0,255,0), 2)
    cv2.imshow("Camera",frame)

    key = cv2.waitKey(1) & 0xFF
    flagOld = flag

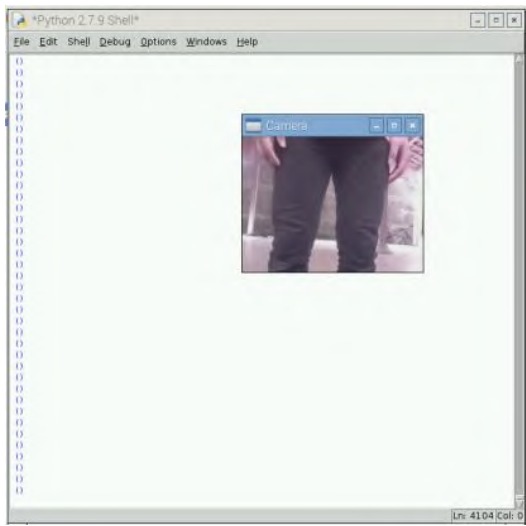
try:
    print "PIR Sensor is working!!";
    print "Ready to sense..";
    while True:
        if GPIO.input(4)==1 and
previous==0:
            print "Gerakan Terdeteksi";
            previous=1;
            startcamera();
            cv2.destroyAllWindows()
            elif GPIO.input(4)==0 and
previous==1:
            print "Sensor PIR siap";
            previous=0;
            time.sleep(0.01);
except KeyboardInterrupt:
    print "QUIT";
    GPIO.cleanup();

```

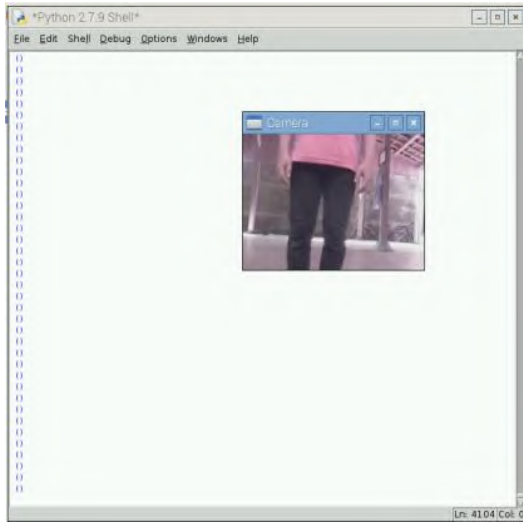
**Gambar B.1 Fungsi gabungan seluruh program**



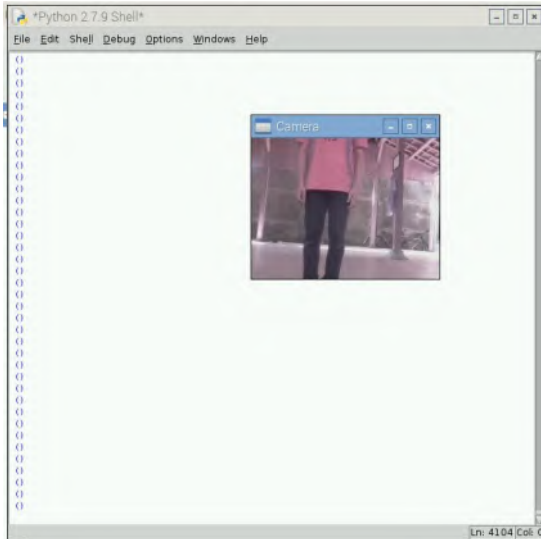
C. Data Gambar Pengujian



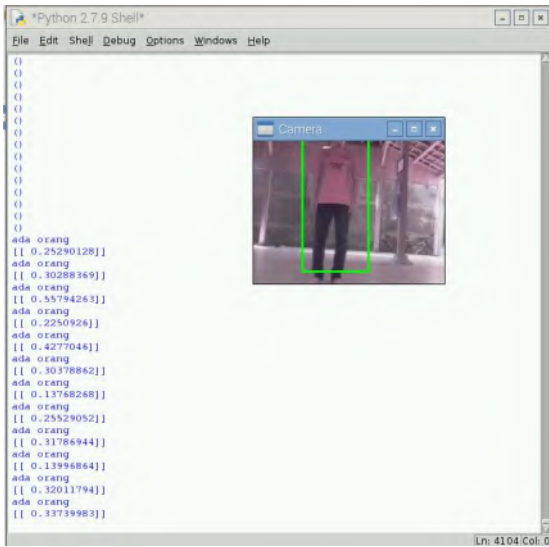
Gambar C.1 Uji Coba HOG Jarak 0.5 Meter



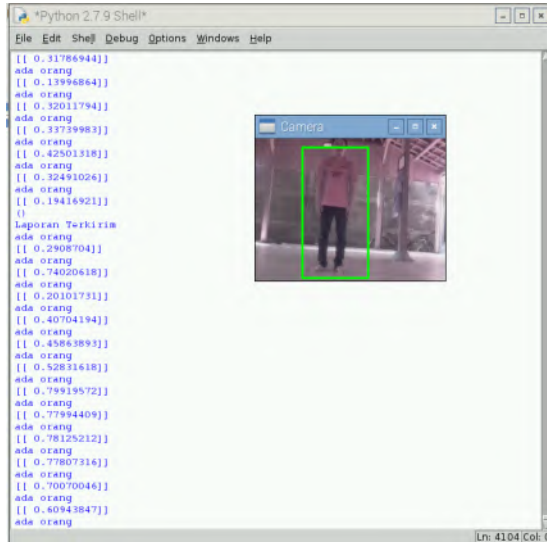
Gambar C.2 Uji Coba HOG Jarak 1 Meter



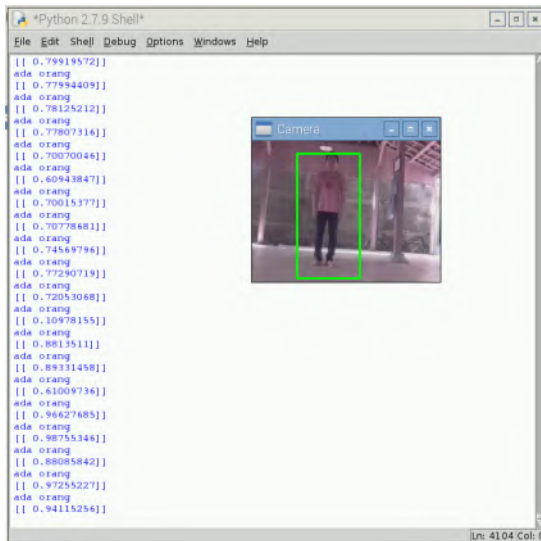
**Gambar C.3 Uji Coba HOG Jarak 1.5 Meter**



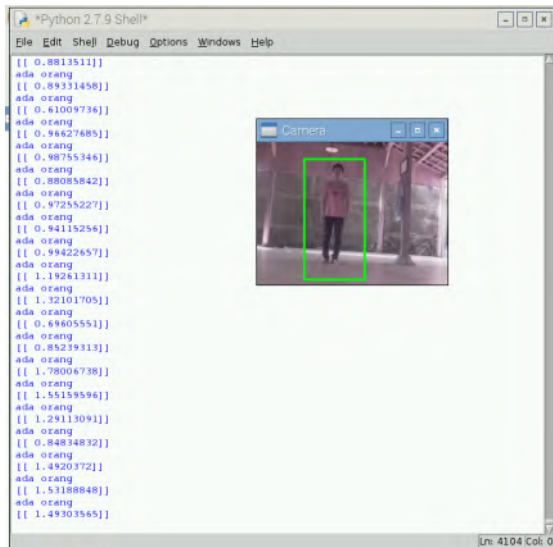
**Gambar C.4 Uji Coba HOG Jarak 2 Meter**



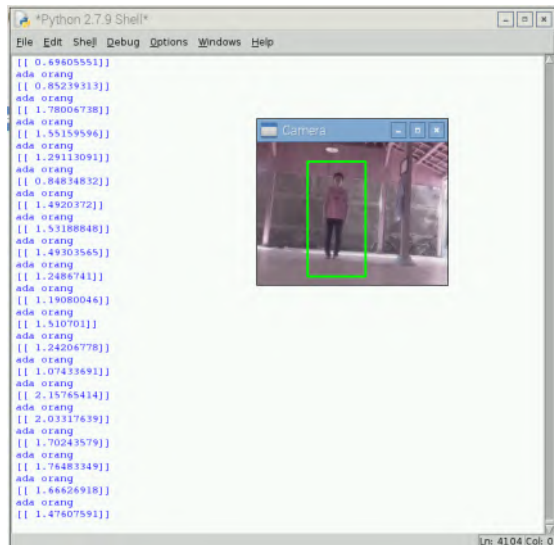
**Gambar C.5 Uji Coba HOG Jarak 2.5 Meter**



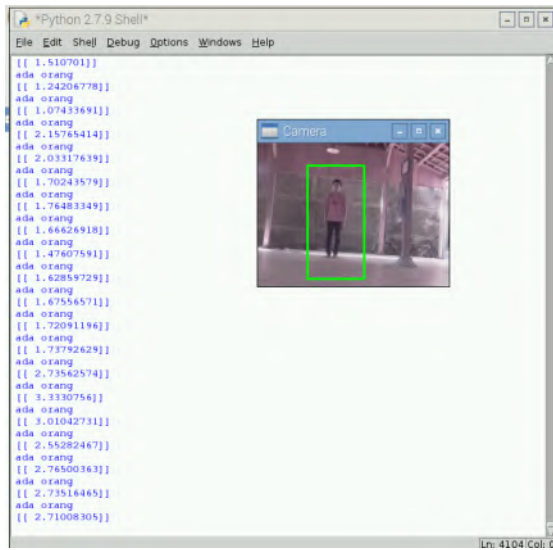
**Gambar C.6 Uji Coba HOG Jarak 3 Meter**



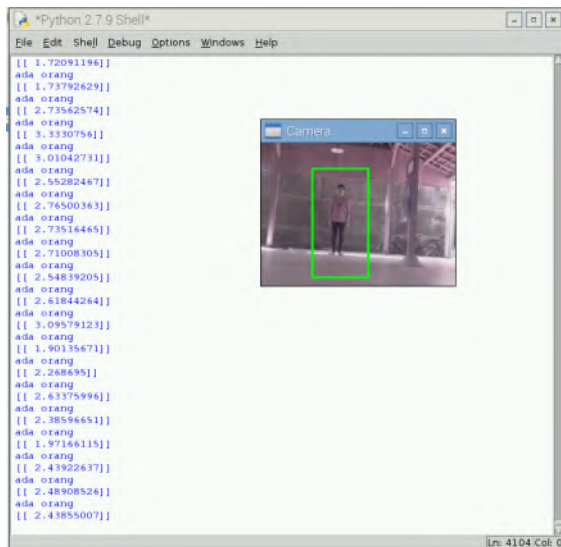
**Gambar C.7 Uji Coba HOG Jarak 3.5 Meter**



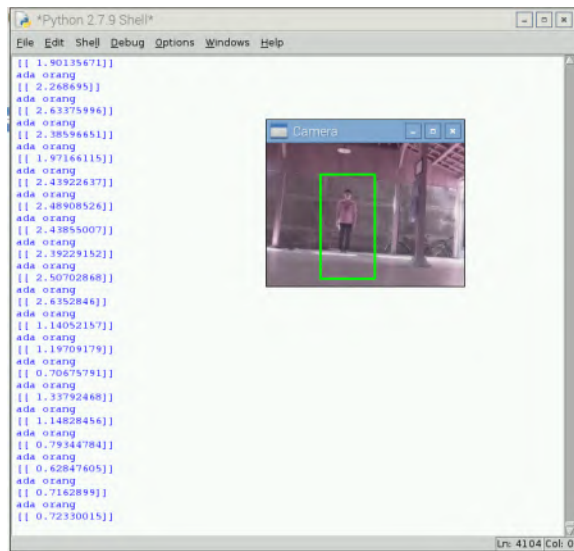
**Gambar C.8 Uji Coba HOG Jarak 4 Meter**



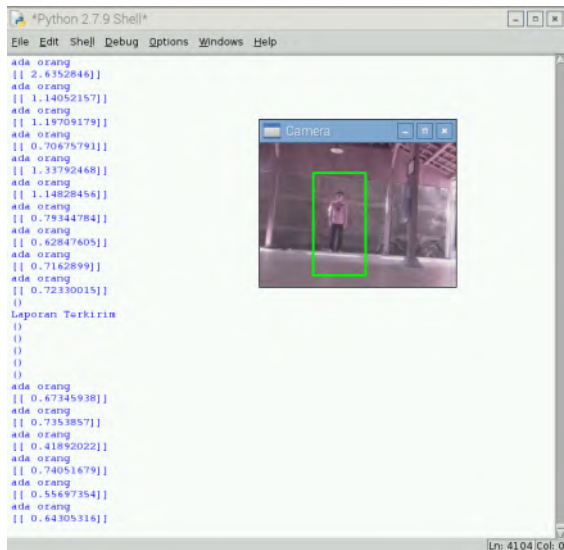
**Gambar C.9 Uji Coba HOG Jarak 4.5 Meter**



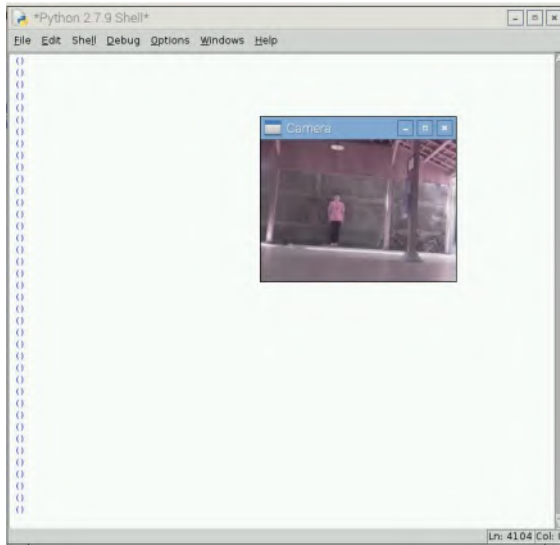
**Gambar C.10 Uji Coba HOG Jarak 5 Meter**



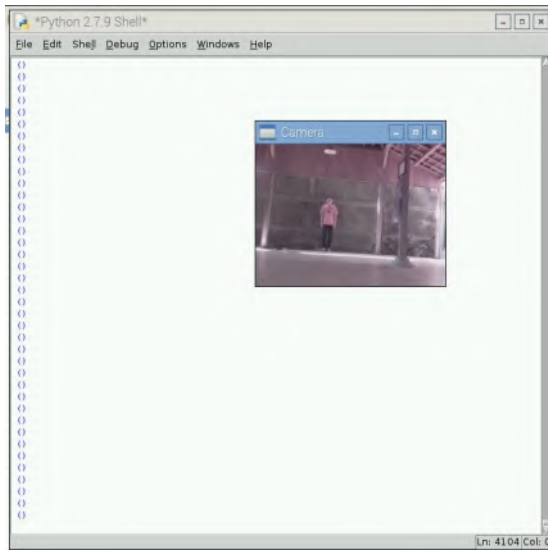
**Gambar C.11 Uji Coba HOG Jarak 5.5 Meter**



**Gambar C.12 Uji Coba HOG Jarak 6 Meter**



**Gambar C.13 Uji Coba HOG Jarak 6.5 Meter**



**Gambar C.14 Uji Coba HOG Jarak 7 Meter**

## **BAB V**

### **KESIMPULAN DAN SARAN**

Bab ini membahas mengenai kesimpulan yang dapat diambil dari hasil uji coba yang telah dilakukan sebagai jawaban dari rumusan masalah yang dikemukakan. Selain kesimpulan, juga terdapat saran yang ditujukan untuk pengembangan penelitian lebih lanjut.

#### **5.1 Kesimpulan**

Dari hasil uji coba yang telah dilakukan terhadap perangkat sistem pengawasan pondok pesantren menggunakan perangkat Raspberry Pi dapat disimpulkan bahwa perangkat sistem ini dapat mendeteksi gerakan manusia yang bergerak hingga berjarak 6 meter dan objek manusia dapat terdeteksi oleh algoritma HOG pada jarak antar 2 meter hingga 6 meter, idealnya alat ini diletakkan 3 meter dari objek atau sasaran tempat pelanggaran santri. Perangkat masih dapat mendeteksi manusia meskipun dengan penambahan sedikit *noise* berupa pemakaian atribut ala santri pondok pesantren.

#### **5.2 Saran**

Saran yang diberikan untuk pengembangan sistem pengawasan batas pondok pesantren yaitu Penggunaan modem pada koneksi sistem ini, karena apabila wifi mati sistem tetap berjalan semestinya. Mengembangkan lagi kinerja dari program dan perbaikan dari segi *casing*.



## DAFTAR PUSTAKA

- [1] “Raspberry Pi: FAQs,”. Available:  
<http://www.raspberrypi.org/faqs#introWhatIs>. [Diakses 11 Maret 2016].
- [2] “Raspberry Pi GPIO,”. Available:  
<http://raspberrypiwebserver.com/gpio/gpio.html>. [Diakses 12 mei 2016].
- [3] “HC-SR501 Human Sensor Module Pyroelectric Infrared,” . Available:  
[http://www.icstation.com/product\\_info.php?products\\_id=1390#UxV\\_FvmSy3B](http://www.icstation.com/product_info.php?products_id=1390#UxV_FvmSy3B). [Diakses 2 Maret 2016].
- [4] “5MP Raspberry Pi Camera Board NoIR,” . Available:  
<https://www.modmypi.com/raspberry-pi-noir-camera-board>. [Diakses 1 Maret 2016].
- [5] “Raspbian OS,” . Available: <http://www.raspbian.org/>. [Diakses mei 1 2016].
- [6] “SMTP,” . Available:  
[http://www.tutorialspoint.com/python/python\\_sending\\_email.htm](http://www.tutorialspoint.com/python/python_sending_email.htm). [Diakses 10 mei 2016].
- [7] “OpenCV,” . Available:  
<http://docs.opencv.org/modules/core/doc/intro.html>. [Diakses 13 mei 2016].
- [8] “Object Detection,” . Available:  
[http://docs.opencv.org/modules/gpu/doc/object\\_detection.html#gpu-hogdescriptor-detectmultiscale](http://docs.opencv.org/modules/gpu/doc/object_detection.html#gpu-hogdescriptor-detectmultiscale). [Diakses juni 2016].
- [9] “Ultrasonic Ranging Detector Mod HC-SR04 Distance Sensor,” . Available: <http://www.sainsmart.com/ultrasonic-ranging-detector-mod-hc-sr04-distance-sensor.html>. [Diakses 1 Maret 2016].
- [10] “Sysstat,” . Available: <http://sebastien.godard.pagesperso-orange.fr/documentation.html>. [Diakses juli 2016].
- [11] “Library geopy,” . Available:

- <http://geopy.readthedocs.org/en/latest/>. [Diakses Mei 2016].
- [12] “picamera,” . Available:  
<http://picamera.readthedocs.org/en/release-1.3/>. [Diakses Mei 2016].
- [13] “HOG,” . Available:  
[http://docs.opencv.org/modules/gpu/doc/object\\_detection.html#gpu-hogdescriptor-detectmultiscale](http://docs.opencv.org/modules/gpu/doc/object_detection.html#gpu-hogdescriptor-detectmultiscale). [Diakses Mei 2016].
- [14] “library MIME,” . Available:  
<https://docs.python.org/2/library/email.html>. [Diakses Mei 2016].

## BIODATA PENULIS



Nur Ahmad Syahid, lahir di Benteng, pada tanggal 22 April 1994. Penulis menempuh pendidikan mulai dari SD Inpres Paccerakkang (2000-2006), SMP Pondok Pesantren IMMIM Putra (2006-2009), MA Pondok Pesantren IMMIM Putra (2009-2012) dan melanjutkan studi di S-1 Teknik Elektro ITS (2012-2016). Selama masa kuliah, penulis aktif dalam banyak organisasi intra maupun ekstra diantaranya

organisasi CSSMoRA ITS yang merupakan organisasi penerima Program Beasiswa Santri Berprestasi (PBSB) oleh kementrian agama, organisasi LDJ dan LDK JMMI ITS, Workshop Teknik Elektro ITS, UKM dan banyak lagi lainnya . Penulis juga menjadi anggota lab B402. Penulis juga menghabiskan masa perkuliahan dengan menyalurkan hobi penulis seperti futsal, bermain dota, *travelling* dan berbagai berolahraga lainnya.

Selama kuliah di Teknik Elektro ITS penulis mengambil bidang minat elektronika. Komunikasi dengan penulis melalui: [nurahmadsyahid@gmail.com](mailto:nurahmadsyahid@gmail.com).